

UNIVERSIDADE FEDERAL DO PARANÁ

MAICON ALEXANDER ILLES FANG

MBEST PRODUCTS – PRODUTOS EM PROMOÇÃO APRESENTADOS EM
DISPOSITIVOS MÓVEIS

CURITIBA

2014

MAICON ALEXANDER ILLES FANG

MBEST PRODUCTS – PRODUTOS EM PROMOÇÃO APRESENTADOS EM
DISPOSITIVOS MÓVEIS

Trabalho de conclusão de curso apresentado como requisito à conclusão do Curso de Tecnologia em Análise e Desenvolvimento de Sistemas, Setor de Educação Profissional e Tecnológica, Universidade Federal do Paraná.

Orientador: Professor Jaime Wojciechowski

CURITIBA

2014

TERMO DE APROVAÇÃO

MAICON ALEXANDER ILLES FANG

**MBEST PRODUCTS – PRODUTOS EM PROMOÇÃO APRESENTADOS EM
DISPOSITIVOS MÓVEIS**

Monografia apresentada como requisito parcial para a obtenção da titulação de especialista, pelo Curso de Pós-Graduação Lato Sensu em Engenharia de Software, da Universidade Federal do Paraná.



Orientador: Professor Jaime Wojciechowski

Curitiba, 26 de Janeiro de 2015

AGRADECIMENTOS

Ao meu orientador Prof. Dr. Jaime Wojciechowski que dedicou o seu tempo me auxiliando na elaboração deste trabalho.

Aos meus familiares e amigos por todo o apoio e compreensão.

À minha noiva por ter me apoiado em todos os momentos e por ter despendido o seu tempo em prol da revisão deste trabalho.

À um amigo em especial chamado Gemerson Ribas pela ajuda, apoio e compreensão durante a realização deste projeto.

À Deus por me conceder a oportunidade de buscar novos horizontes, de colocar pedras em meu caminho e por me fazer capaz de supera-las.

RESUMO

Com o crescente avanço de novas tecnologias e a popularização da internet, assim como de smartphones, as pessoas estão cada vez mais conectadas a rede de computadores. A atual rede de computadores facilita a vida das pessoas tanto através da troca de mensagens via redes sociais, assim como deixando mais fácil o acesso a pesquisas, entretenimento, informações, entre outros itens. Tendo em vista este cenário, é crescente a oferta de softwares que vem ganhando espaço no mercado. A proposta principal do software apresentado no presente trabalho é proporcionar ao usuário a fácil visualização de produtos em promoção através de um aplicativo em seu dispositivo móvel. Uma característica importante deste produto é que além de tornar possível a visualização de produtos em promoção de uma empresa, este também fornece a localização exata da mesma. Além disso, este software apresenta filtros no dispositivo o que possibilita diferentes formas de consulta aos produtos.

Palavras-chaves: Aplicativo para dispositivos móveis. Produtos em promoção. Software.

ABSTRACT

Considering the advance of new technologies as well as the popularization of internet and smartphones, people have become even more connected to computer network. The current computer network helps to make peoples' life by allowing people to exchange messages through social networks and by making easier to do a research, to find information or entertainment, among others. Seeing this scenario, the quantity of software which is getting importance in the market has increased. The main proposal of the software presented here is providing to the user the facility of visualizing products on sale through a mobile app. Another characteristic of this software is that the user not only can visualize products on sale in a specific business but also can visualize the business location. In addition, this software presents filters which allow searching products through different ways.

Keywords: Mobile App. Products on sale. Software.

LISTA DE FIGURAS

FIGURA 1- MOTOROLA DYNATAC 8000X	13
FIGURA 2- AS EMPRESAS QUE FORMAM A ALIANÇA – OHA	15
FIGURA 3- ARQUITETURA ANDROID	16
FIGURA 4- APLICATIVO CASAS BAHIA	17
FIGURA 5- APLICATIVO SUBMARINO	18
FIGURA 6- APLICATIVO RENNER	19
FIGURA 7- ESTRUTURA DO WEBSERVICE	20
FIGURA 8- WBS FASES DO PROJETO	22
FIGURA 9- GRÁFICO DE GANTT	23
FIGURA 10- TELA INICIAL EMPRESA, EMPRESA PRODUTO, PRODUTO (ATUALIZAR)	27
FIGURA 11- TELA DE LOGIN DO SISTEMA WEB	28
FIGURA 12- TELA PRINCIPAL COM OS MENUS DO SISTEMA WEB	28
FIGURA 13- TELA DE CADASTRO DE USUÁRIO DO SISTEMA WEB	28
FIGURA 14- TELA DE CADASTRO DE PAIS DO SISTEMA WEB	29
FIGURA 15- TELA DE CADASTRO DE ESTADO DO SISTEMA WEB	29
FIGURA 16- TELA DE CADASTRO DE CIDADE DO SISTEMA WEB	30
FIGURA 17- TELA DE CADASTRO DE PERFIL DO SISTEMA WEB	30
FIGURA 18- TELA DE CADASTRO DE ATIVIDADE DO SISTEMA WEB	30
FIGURA 19- TELA DE CADASTRO DE CATEGORIA DO SISTEMA WEB	31
FIGURA 20- TELA DE CADASTRO DE EMPRESA DO SISTEMA WEB	31
FIGURA 21- TELA DE CADASTRO DE FUNCIONÁRIO DO SISTEMA WEB	32
FIGURA 22- TELA DE CADASTRO DE PRODUTO DO SISTEMA WEB	33

LISTA DE SIGLAS

- ADT - Android Development Tools (Ferramentas de desenvolvimento Android).
- IDE - Integrated Development Environment (Ambiente Integrado de Desenvolvimento).
- WEB - World Wide Web (Teia Mundial)
- JSF - Java Server Faces (Java é uma linguagem de programação orientada a objeto)
- WBS - Work Breakdown Structure (Estrutura analítica do projeto)
- TCC - Trabalho de conclusão de curso
- UC - Use Case (Uso de caso)
- UML - Unified Modeling Language (Linguagem Unificada de Modelagem)
- EAP - Estrutura Analítica do Projeto

SUMÁRIO

1 INTRODUÇÃO	11
1.1 OBJETIVO DO PROJETO	11
1.1.1 Objetivo Geral	11
1.1.2 Objetivos Específicos	11
1.2 JUSTIFICATIVA	12
1.3 ESTRUTURA DO TRABALHO	12
2 FUNDAMENTAÇÃO TEÓRICA	13
2.1 SMARTPHONE OU TELEFONE INTELIGENTE	13
2.2 ANDROID	14
2.2.1 Plataforma Android	15
2.2.1 Vantagens em Desenvolver em Android	17
2.3 TRABALHOS RELACIONADOS	17
2.4 PROPAGANDA	19
2.5 SISTEMAS WEB SERVICES E SUA INTEGRAÇÃO COM APLICAÇÕES ANDROIDS	19
3 METODOLOGIA	21
3.1 VISÃO GERAL DO SISTEMA	21
3.2 UML LINGUAGEM UNIFICADA DE MODELAGEM	21
3.3 WBS E GRAFICO DE GANTT	22
3.4 PLANO DE RISCO	23
3.5 RESPONSABILIDADES	24
3.6 MATERIAIS	24
3.6.1 Hardware	24
3.6.2 Software	24
3.6.3 Infraestrutura	25
3.7 DESENVOLVIMENTO DO PROJETO	25
3.7.1 Descrição do Escopo	25
4 APRESENTAÇÃO DO PROJETO	26
4.1 INSTALAÇÃO	27
4.2 PROTOTIPOS DE TELAS	27
4.2.1 Sistema Android	27

4.2.2 Sistema Web	27
5 CONSIDERAÇÕES FINAIS	33
REFERÊNCIAS	34
APÊNDICES	36

1 INTRODUÇÃO

A finalidade deste documento é coletar, analisar e definir as necessidades e características de nível superior do mBestProducts. Este documento relaciona os recursos de que os usuários e envolvidos no problema necessitam para melhorar a forma de busca de informações. Os detalhes de como o mBestProducts atende a essas demandas estão descritos nas especificações suplementares e de caso de uso.

1.1. OBJETIVO DO PROJETO

Esta seção apresenta os objetivos, tanto geral como específicos do projeto, buscando estabelecer o que se espera conquistar com a finalização do desenvolvimento do mesmo.

1.1.1. Objetivo Geral

Desenvolver um aplicativo que apresente ao/s usuário/s de forma transparente, simples e objetiva os produtos em promoção.

1.1.2. Objetivos Específicos

Desenvolver somente para dispositivos móveis com sistema operacional android.

Exibir somente produtos que estejam com a data de promoção igual o superior a data atual/corrente.

Somente um usuário terá permissão para cadastrar os dados no sistema.

1.2. JUSTIFICATIVA

Analisar a utilização do aplicativo mBestProducts no mercado.

Atualmente as aplicações existentes no mercado não possuem um foco somente na exibição de produtos em promoção.

O projeto tem como principal diferencial a objetividade e agilidade na apresentação dos produtos em promoção.

1.3. ESTRUTURA DO TRABALHO

Os próximos capítulos deste documento pretendem apresentar o desenvolvimento deste trabalho. Este documento está dividido da seguinte forma, o Capítulo 2 apresenta a fundamentação teórica e uma breve revisão da literatura, com o intuito de posicionar melhor o leitor em relação a este trabalho. No Capítulo 3 encontra-se a explicação de como o sistema foi modelado, os recursos e ferramentas utilizadas para este fim, como o EAP (Estrutura Analítica do Projeto) e UML (Linguagem de Modelagem Unificada). É apresentado também neste capítulo, o gráfico de Gantt, o plano de atividades, o plano de riscos e os recursos de hardware e software. No Capítulo 4 apresenta-se o sistema, visando mostrar principalmente a documentação de telas e suas funcionalidades. Finalmente, o

Capítulo 5 encerra o documento concluindo o trabalho. Este é seguido pela bibliografia e a seção de apêndices.

2 FUNDAMENTAÇÃO TEÓRIA

Este capítulo tem por objetivo apresentar uma introdução ao smartphone, android, também explicar o funcionamento do sistema e alguns conceitos utilizados, sua utilização em algumas áreas chaves, será apresentado também o que são sistemas web e sua integração com sistemas android.

2.1 SMARTPHONE OU TELEFONE INTELIGENTE

A palavra smartphone é uma evolução do celular que teve seu início em 1983 um aparelho chamado Motorola DynaTac 8000x, pesando aproximadamente um (1) quilo, entre as características do aparelho estava a duração da bateria que permitia um conversa de no máximo 30 minutos, o tamanho do aparelho, não tinha comunicação com a internet e sem mencionar o custo da aquisição (SOUZA, 2011).



Figura 1. Motorola DynaTac 8000x

Fonte: <http://oldschool.tblog.com/post/1970189498>

Avanços significativos ocorreram, aumento na duração da bateria, diminuição do preço, maior cobertura da rede, aumentando cada vez o número de

celular no mercado, o surgimento da internet que fez com que usuários solicitassem por novas funcionalidades, aplicativos e maior interação com o celular (SOUZA, 2011).

Com a junção dos avanços ocorridos, algumas plataformas surgiram no mercado como, o Android, Windows Phone, Symbian, entre outros (SOUZA, 2011).

Algumas empresas denominam o smartphone como telefone inteligente, no entanto não existe uma definição padronizada sobre o nome, pois o fato é que algumas características mudaram muito desde sua concepção, passando de agenda de compromisso, lista de contato para outro patamar sendo possível hoje acessar a rede de computadores, interagir com outras pessoas via redes sociais, e também teve sua propagação expandida para outras áreas como por exemplo: saúde, educação, esporte, lazer, entre muitas outras (RODRIGUES, 2009).

2.2 ANDROID

Inicialmente foi desenvolvido por uma startup americana do Vale do Silício chamada Android Inc. porém em 2005 foi comprado pela empresa Google, mas somente em 2007 iniciou sua popularização no mercado, tornando-o uma plataforma *open source* focado no desenvolvimento para smartphone (GONÇALVES, 2011).

Open Handset Alliance (OHA) juntamente com a Google formam as 40 empresas que atualmente mantem o Android. Entre algumas empresas conhecidas no mercado podemos citar a Intel, Motorola, Samsung, entre outras de diferente ramos de atividades (GONÇALVES, 2011).

Na imagem abaixo é possível visualizar as empresas que compõe a Aliança (OHA) que padronizam o Android.



Figura 2. As empresas que formam a Aliança – OHA

Fonte: <http://fema.edu.br/images/arqTccs/0911270073.pdf>

2.2.1 Plataforma Android

A figura a baixo exhibe os elementos que compõe o sistema operacional android.

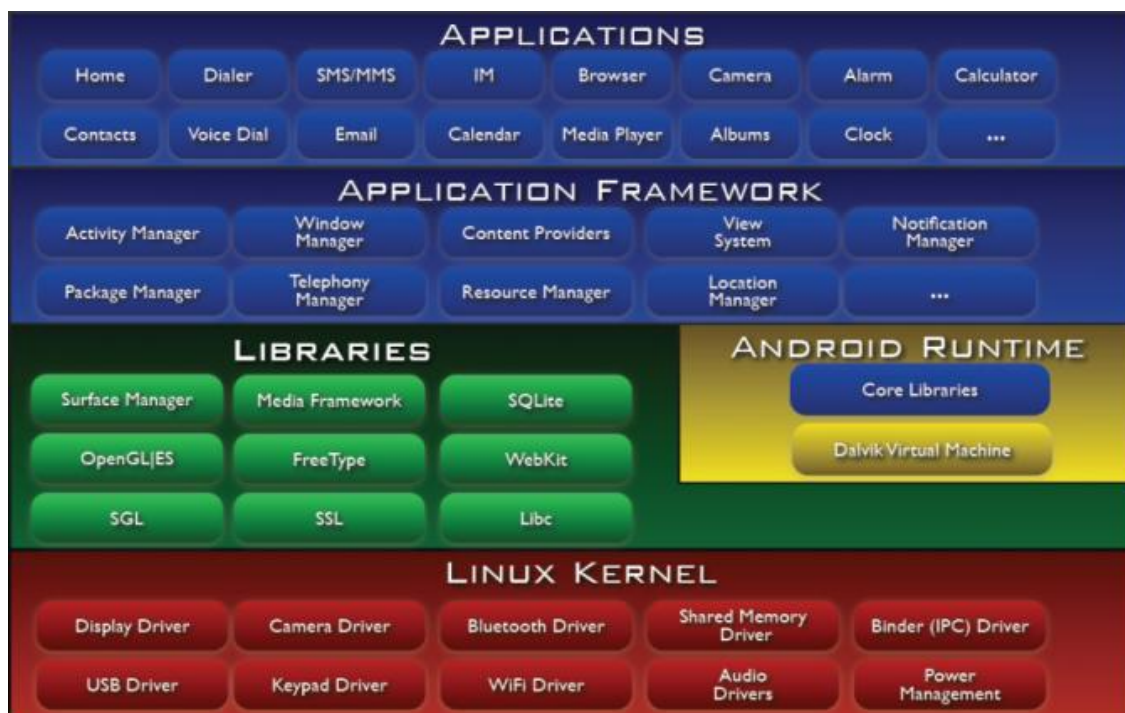


Figura 3. Arquitetura Android

Fonte: <http://www.sergioprado.org/wp-content/uploads/system-architecture.jpg>

O Android é um sistema operacional construído em cima de outro sistema operacional *open source* chamando Linux, permitindo assim a interação entre o hardware e o software e também dando suporte aos principais serviços do sistema como gerenciamento de processos e de memória (MARTINS, 2009).

Foi desenvolvido na linguagem de programação Java, porém não é executado em uma máquina virtual Java tradicional, e sim outra chamada *Dalvik* que voltada reformulada para dispositivos móveis (MARTINS, 2009).

A figura acima também exhibe as bibliotecas feitas em C/C++ que são utilizadas por desenvolvedores para criação de novos aplicativos, que envolvem imagem, vídeo, áudio, gps, navegação na internet, comunicação com banco de dados, entre outras (NOGUEIRA, 2012).

Runtime do Android contém as bibliotecas da linguagem de programação Java assim fornecendo a maior parte das funcionalidades (NOGUEIRA, 2012).

Applications (Aplicativos) o sistema operacional android por padrão fornece aplicativos básicos que incluem, calendário, e-mail, navegação a internet, câmera, contatos, entre muitos outros (NOGUEIRA, 2012).

Application Framework (Bloco de aplicação) onde se encontram as APIs do Android, geralmente, são aplicações que executam na plataforma possibilitando aos desenvolvedores criarem robustas aplicações (NOGUEIRA, 2012).

2.2.1 Vantagens em Desenvolver em Android

Desenvolver para sistemas operacional android possibilita a criação de aplicativos criativos, úteis, inteligentes, interativos, robustos, simples, confiáveis. O sistema operacional é um dos mais utilizados por dispositivos moveis, de código aberto e cada vez ganha mais mercado.

2.3 TRABALHOS RELACIONADOS

Casas Bahia

É um aplicativo gratuito desenvolvido pela loja casas bahia, oferece uma lista de produtos e promoções presentes no próprio site da loja, clientes cadastrados podem fazer compras, o aplicativo também direciona o usuário a loja mais próxima (ALINE JESUS, 2014).

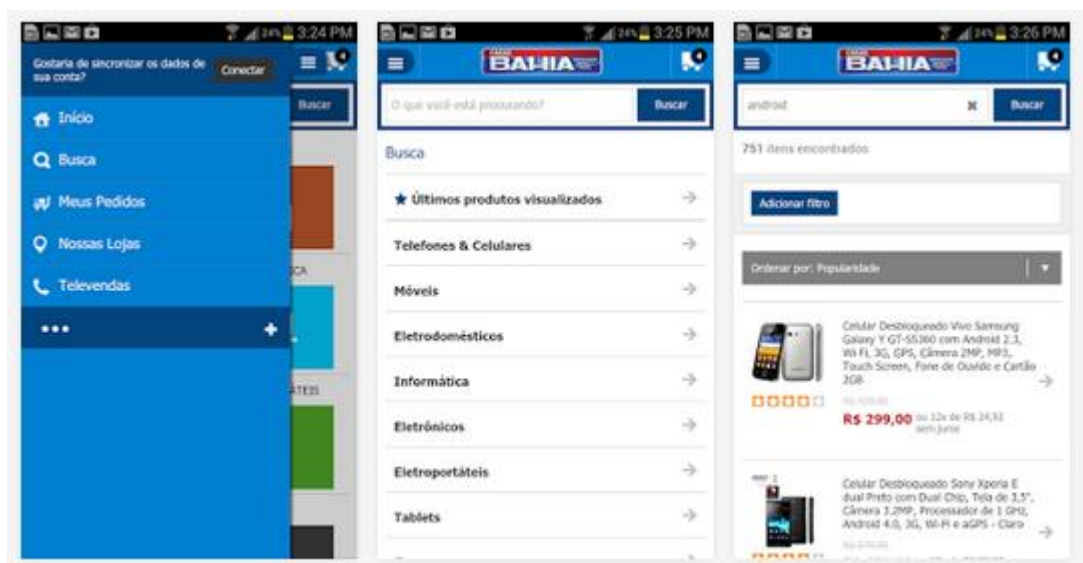


Figura 4. Aplicativo Casas Bahia

Fonte:

https://play.google.com/store/apps/details?id=com.novapontocom.casasbahia&hl=pt_BR

Submarino

Aplicativo gratuito que apresenta produtos e promoções e onde é possível realizar uma ou mais compras, com o leitor de código de barras é possível comparar preços, também é possível marcar um ou mais produtos como favorito (B2W DIGITAL, 2014).

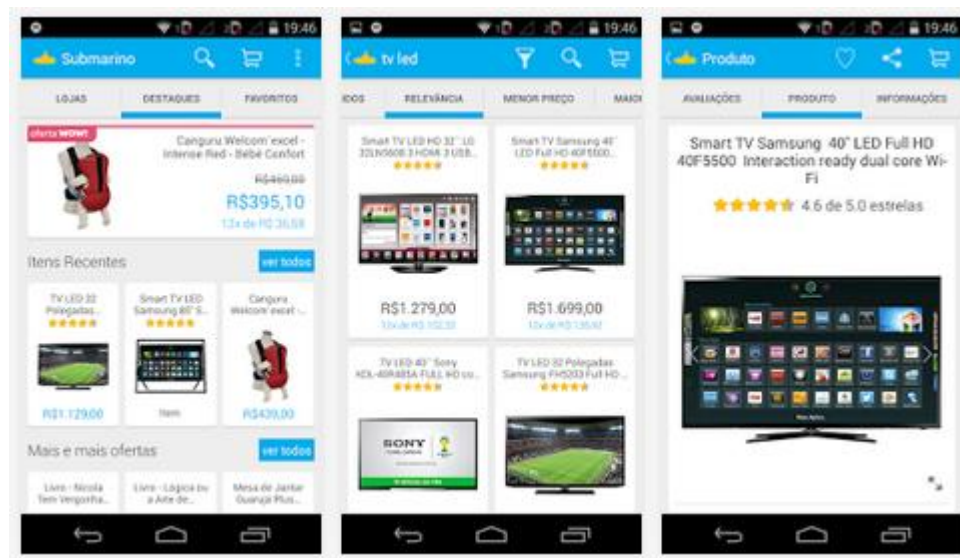


Figura 5. Aplicativo Submarino

Fonte: https://play.google.com/store/apps/details?id=com.b2w.submarino&hl=pt_BR

Renner

Aplicativo exibe na tela do disposto móvel produtos com ou sem promoções, é possível realizar uma ou mais compras, conta com a funcionalidade de lembranças as mulheres de seus compromissos que estão na agenda, também exibe quais as lojas renner estão mais próximas (LOJAS RENNER, 2014).

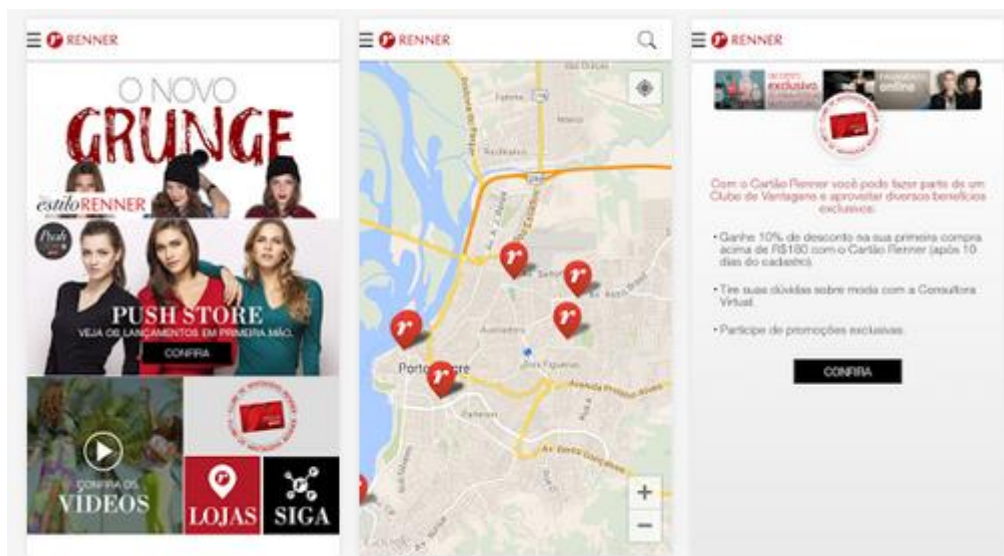


Figura 6. Aplicativo Renner

Fonte: https://play.google.com/store/apps/details?id=br.com.lojasrenner&hl=pt_BR

2.4 PROPAGANDA

A propaganda busca divulgar um produto e podendo alavancar sua imagem tornando o mais aceito no mercado, através de meio de comunicação como TV por exemplo que pode se custar mais caro dependendo do horário, tempo ou emissora em que o produto será exibido, também pode ser divulgado por rádio, jornais impressos, panfleto, boca-a-boca, redes sociais entre tantos outros (SILVA, 2011).

Vários termos para a propaganda existe no mercado entre elas estão: propaganda de produto, serviço, marca, institucional, persuasiva, comparativa, proteção, corretiva, lembrança e outros (SILVA, 2011).

2.5 SISTEMAS WEB SERVICES E SUA INTEGRAÇÃO COM APLICAÇÕES ANDROIDS

Sistemas web services são serviços oferecidos por outros sistemas que podem ser postados, localizados e chamados via web, estes podem ser construídos

para simples tarefas como, por exemplo, envio e recebimento de arquivos ou para funções mais complexas como transações bancárias. Além disso, podem ser utilizados também para compor um ou mais módulos de um sistema comercial ou industrial (TAME, LIMA, 2005).

A comunicação é feita via protocolos e não depende de plataforma SOAP, XML e HTTP, é acessível via mensagens XML, ou seja, em formato texto, que contém o formato da mensagem, localização e o tipo das informações, um cliente faz a requisição do que deseja ao servidor que por sua vez devolve em formato mensagem o que foi requerido (TAME, LIMA, 2005).



Figura 7. Estrutura do Webservice

Fonte: toolsweb, O que é Web Services? -

<http://www.toolsweb.com.br/?webservice>

Para o envio dados para a aplicação Android foi criado um web service utilizando uma biblioteca chamada RestEasy, para envio de dados do software web para o aplicativo Android foi incorporado ao projeto o JSON (JavaScript Object Notation) é utilizado para troca de dados, informações entre sistemas. Sua adoção deve-se por ser um formato fácil de ler comparado a outros formatos como XML, HTML, entre outros.

3. METODOLOGIA

Este capítulo apresenta a metodologia de como foi feito o trabalho, contemplando modelagem do sistema, ferramentas, recursos para criar o projeto, gráfico de Gantt, o plano de atividades, o plano de riscos, plano de testes, hardware software.

3.1 VISAO GERAL DO SISTEMA

O projeto mBestProducts tem por finalidade apresentar os produtos de uma ou mais lojas que estão em promoção na tela do dispositivo, o processo é feito com um usuário/funcionário habilitado cadastrando um ou mais produtos no sistema web de uma determinada loja, após realizar o cadastro corretamente, é enviado web service um arquivo em formato JSON os produtos para os sistema android que faz o processamento do arquivo e exibe os produtos ao usuário.

3.2 UML LINGUAGEM UNIFICADA DE MODELAGEM

Para construir o sistema também foram utilizados documentos fornecidos pela UML (Unified Modeling Language) traduzindo para o português significa Linguagem Unificada de Modelagem. Seu benefício troou-se possível ter uma visão macro e geral do projeto, entendimento dos atores envolvidos, interação entre as tabelas e classes do sistema.

3.3 WBS E GRAFICO DE GANTT

A WBS Work Braskdown Structure ou EAP Estrutura Analítica do Projeto, seu objetivo é dividir as atividades em outras menores ficando assim mais fácil de gerenciar desde as genéricas até as mais específicas (INSTITUTE, 2008).

As atividades de projeto foram divididos em cinco fases principais que são: gerenciamento do projeto, iniciação, elaboração, construção e transição que podem ser vistas na figura WBS.

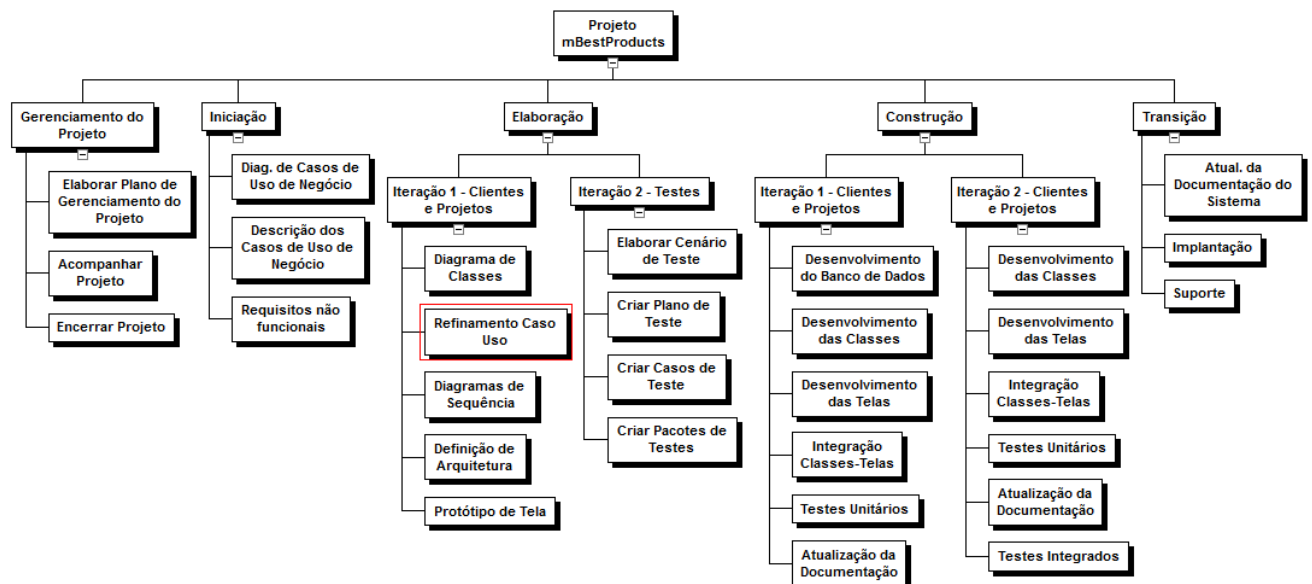


Figura 8. WBS fases do projeto

A imagem abaixo está o gráfico de Gantt que representa as atividades do processo de desenvolvimento do software em relação ao tempo de elaboração.

		Nome	Custo	Duração	Trabalho	Início	Fim	Nome dos Recursos
1		LmbestProducts	R\$ 15800,00	51,25 dias?	395 horas	10/10/13 08:00	20/12/13 10:00	
2		Gerenciamento do Projeto	R\$ 6400,00	47,75 dias?	160 horas	10/10/13 08:00	16/12/13 15:00	
3		Elaborar Plano de Gerenciamento do Projeto	R\$ 320,00	1 dia?	8 horas	10/10/13 08:00	10/10/13 17:00	Gerente de Projetos
4		Acompanhar Projeto	R\$ 6000,00	18,75 dias?	150 horas	20/11/13 08:00	16/12/13 15:00	Gerente de Projetos
5		Encerrar Projeto	R\$ 80,00	0,25 dias?	2 horas	12/12/13 08:00	12/12/13 10:00	Gerente de Projetos
6		Inicição	R\$ 960,00	2 dias?	24 horas	15/10/13 08:00	16/10/13 17:00	
7		Diag. de Casos de Uso de Negócio	R\$ 320,00	1 dia?	8 horas	15/10/13 08:00	15/10/13 17:00	Gerente de Projetos
8		Descrição dos Casos de Uso de Negócio	R\$ 320,00	1 dia?	8 horas	16/10/13 08:00	16/10/13 17:00	Gerente de Projetos
9		Requisitos não funcionais	R\$ 320,00	1 dia?	8 horas	16/10/13 08:00	16/10/13 17:00	Gerente de Projetos
10		Elaboração	R\$ 1040,00	28,125 dias?	26 horas	12/10/13 08:00	21/11/13 09:00	
11		Iteração 1 - Clientes e Projetos	R\$ 720,00	27,5 dias?	18 horas	12/10/13 08:00	20/11/13 13:00	
12		Diagrama de Classes	R\$ 160,00	0,5 dias?	4 horas	12/10/13 08:00	14/10/13 13:00	Gerente de Projetos
13		Refinamento Caso Uso	R\$ 160,00	0,5 dias?	4 horas	18/10/13 08:00	18/10/13 13:00	Gerente de Projetos
14		Diagramas de Sequência	R\$ 160,00	0,5 dias?	4 horas	20/10/13 08:00	21/10/13 13:00	Gerente de Projetos
15		Definição de Arquitetura	R\$ 80,00	0,25 dias?	2 horas	22/10/13 08:00	22/10/13 10:00	Gerente de Projetos
16		Protótipo de Tela	R\$ 160,00	0,5 dias?	4 horas	20/11/13 08:00	20/11/13 13:00	Analista Desenvolvedor
17		Iteração 2 - Testes	R\$ 320,00	1,125 dias?	8 horas	20/11/13 08:00	21/11/13 09:00	
18		Elaborar Cenário de Teste	R\$ 80,00	0,25 dias?	2 horas	20/11/13 08:00	20/11/13 10:00	Testador
19		Criar Plano de Teste	R\$ 40,00	0,125 dias?	1 hora	20/11/13 08:00	20/11/13 09:00	Testador
20		Criar Casos de Teste	R\$ 160,00	0,5 dias?	4 horas	20/11/13 08:00	20/11/13 13:00	Testador
21		Criar Pacotes de Testes	R\$ 40,00	0,125 dias?	1 hora	21/11/13 08:00	21/11/13 09:00	Testador
22		Construção	R\$ 1320,00	4,375 dias?	33 horas	20/11/13 08:00	26/11/13 11:00	
23		Iteração 1 - Clientes e Projetos	R\$ 800,00	3,25 dias?	20 horas	20/11/13 08:00	25/11/13 10:00	
24		Desenvolvimento do Banco de Dados	R\$ 120,00	0,375 dias?	3 horas	20/11/13 08:00	20/11/13 11:00	Analista Desenvolvedor
25		Desenvolvimento das Classes	R\$ 120,00	0,375 dias?	3 horas	21/11/13 08:00	21/11/13 11:00	Analista Desenvolvedor
26		Desenvolvimento das Telas	R\$ 160,00	0,5 dias?	4 horas	21/11/13 08:00	21/11/13 13:00	Analista Desenvolvedor
27		Integração Classes-Telas	R\$ 240,00	0,75 dias?	6 horas	22/11/13 08:00	22/11/13 15:00	Analista Desenvolvedor
28		Testes Unitários	R\$ 80,00	0,25 dias?	2 horas	23/11/13 08:00	25/11/13 10:00	Analista Desenvolvedor
29		Atualização da Documentação	R\$ 80,00	0,25 dias?	2 horas	23/11/13 08:00	25/11/13 10:00	Analista Desenvolvedor
30		Iteração 2 - Clientes e Projetos	R\$ 520,00	1,375 dias?	13 horas	24/11/13 08:00	26/11/13 11:00	
31		Desenvolvimento das Classes	R\$ 80,00	0,25 dias?	2 horas	24/11/13 08:00	25/11/13 10:00	Analista Desenvolvedor
32		Desenvolvimento das Telas	R\$ 80,00	0,25 dias?	2 horas	24/11/13 08:00	25/11/13 10:00	Analista Desenvolvedor
33		Integração Classes-Telas	R\$ 80,00	0,25 dias?	2 horas	24/11/13 08:00	25/11/13 10:00	Analista Desenvolvedor
34		Testes Unitários	R\$ 80,00	0,25 dias?	2 horas	25/11/13 08:00	25/11/13 10:00	Testador
35		Atualização da Documentação	R\$ 120,00	0,375 dias?	3 horas	26/11/13 08:00	26/11/13 11:00	Analista Desenvolvedor
36		Testes Integrados	R\$ 80,00	0,25 dias?	2 horas	26/11/13 08:00	26/11/13 10:00	Testador
37		Transição	R\$ 6080,00	22,25 dias?	152 horas	20/11/13 08:00	20/12/13 10:00	
38		Atual. de Documentação do Sistema	R\$ 80,00	0,25 dias?	2 horas	20/11/13 08:00	20/11/13 10:00	Gerente de Projetos
39		Implantação	R\$ 4000,00	4,167 dias?	100 horas	20/11/13 08:00	26/11/13 09:20	Analista Desenvolvedor ;Te...
40		Suporte	R\$ 2000,00	6,25 dias?	50 horas	12/12/13 08:00	20/12/13 10:00	Analista Desenvolvedor
Gráfico de Gantt - Gestão de Projetos - Final								

Figura 9. Gráfico de GANTT

3.4 PLANO DE RISCO

Para minimizar os possíveis problemas durante o processo de desenvolvimento do software foram levantados alguns riscos e como contorná-los. Sua utilização facilita maior entendimento do projeto, previne alguns erros que podem ocorrer durante sua confecção, aumentando a legibilidade e confiança do projeto. O plano de risco pode ser visualizado no Apêndice E.

3.5 RESPONSABILIDADES

Utilizada para listar as tarefas do projeto que foram implementadas pelo integrante da equipe. Todas as responsabilidades podem ser visualizadas no Apêndice F.

3.6 MATERIAIS

Neste tópico será apresentado o software, hardware e infraestrutura que foram utilizados para desenvolver o projeto.

3.6.1 Hardware

Os equipamentos visto abaixo foram utilizados para o desenvolvimento do projeto.

- Toshiba-L505-S5971 com processador Intel core 2 duo e com 4 GB de memória, com HD de 250 GB.
- Smartfone Sony Xperian E, processador Cortex A5 1000MHz, 2000MB de memória interna.

3.6.2 Software

Neste tópico podem ser vistos os softwares utilizados para desenvolver o projeto:

- Sistema operacional Windows 7 32 bits
- Android versão 4.0

3.6.3 Infraestrutura

Abaixo estão listados os itens que juntos foram a infraestrutura que foi confeccionado o projeto:

- Apache tomcat versão 7
- MySQL Workbench 6.0 CE
- Eclipse Kepler Windows 32 bits
- Astah
- Balsamiq Mockups para Desktop Versão 2.2.9
- Chrome versão 34.0
- DBDesigner 4

3.7 DESENVOLVIMENTO DO PROJETO

Neste tópico serão apresentados a definição do projeto descrevendo de forma macro seu funcionamento, as entregas do projeto, alguns itens que o sistema não contempla e as limitações do projeto.

3.7.1 Descrição do Escopo

- Descrição do escopo do projeto

Para que os produtos possam ser visualizados na tela do celular android, primeiramente um usuário habilitado deve ter cadastrados os produtos corretamente, no smartphone o usuário clica na aplicação que por sua vez exibe as empresas cadastradas no sistema web, ao selecionar uma empresa o sistema exibe os produtos com nome e o valor da promoção, o aplicativo permite ainda que o usuário visualize mais detalhes do produto ao seleciona-lo. Outra funcionalidade do

aplicativo e a exibição dos produtos de acordo com a localização geográfica do usuário que lista os produtos pelas empresas mais próximas.

➤ Entregas do projeto

Macro 1	Declaração do espoco
Macro 2	Proposta do produto
Macro 3	Produto testado
Macro 4	Produto homologado

➤ Não contempla o projeto

- O sistema não exibe produtos que não estejam cadastrados no sistema web
- O sistema não faz integração com outros sistemas
- O sistema envia no e-mail produtos em promoção que são exibidos
- Demais requisitos que não estiverem na documentação não serão implementados

➤ Limitação do projeto

- O projeto foi desenvolvido por um integrante, contando com apoio e ajuda de colegas de trabalho e do meio acadêmico.

4. APRESENTACAO DO PROJETO

Neste capítulo será apresentado o projeto, como fazer sua instalação, os protótipos de telas e os testes realizados.

4.1 INSTALAÇÃO

Alguns passos devem ser feitos antes de ter acesso as funcionalidades do sistema, abaixo são listados os itens.

- 1 – Ter o Apache Tomcat 7 instalado e configurado na máquina.
- 2 – Ter o MySQL instalado e configurado no computador.
- 3 – Ter o Chrome ou Firefox instalado no computador.
- 4 – Ter a aplicação web configurada no computador.
- 5 – Sistema web e dispositivo móvel ter acesso a internet
- 6 – Ter a aplicação instalada no dispositivo móvel

4.2 PROTOTIPOS DE TELAS

Segue abaixo os protótipos de telas do sistema web e do android.

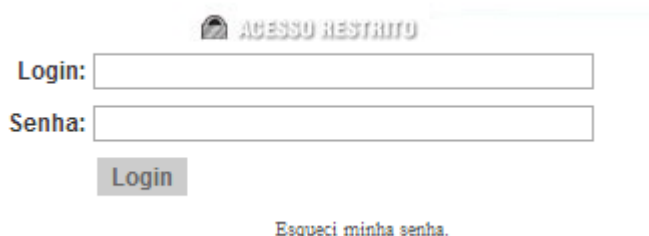
4.2.1 Sistema Android



Figura 10. Tela inicial Empresa, empresa produto, produto (Atualizar)

4.2.2 Sistema Web

Na tela acesso ao sistema o usuário entra com login e senha conforme previamente cadastrado no sistema pelo administrador, dependendo da configuração, nível de acesso, algumas telas podem não serem visualizadas.

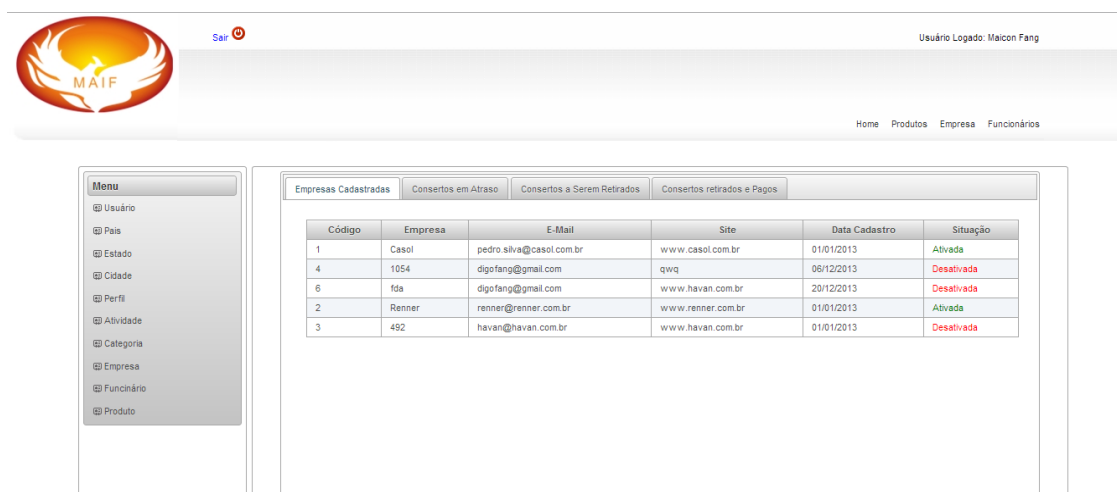


A tela de login do sistema web apresenta o seguinte layout:

- Logo MAIF no canto superior esquerdo.
- Texto "ACESSO RESTRITO" no topo central.
- Campos de entrada para "Login:" e "Senha:".
- Botão "Login" abaixo dos campos.
- Link "Esqueci minha senha." abaixo do botão.

Figura 11. Tela de login do sistema web

Na tela principal do sistema o usuário com perfil administrador poderá navegar entre as telas do sistema, também visualizar, inserir, alterar e excluir qualquer registro, é exibido o usuário que está conectado, as empresa de acordo com sua situação ativo ou inativo.



A tela principal do sistema web apresenta o seguinte layout:

- Logo MAIF no canto superior esquerdo.
- Barra de navegação superior com links: Home, Produtos, Empresa, Funcionários.
- Barra de usuário logado: "Usuário Logado: Maicon Fang".
- Menu lateral com opções:
 - Menu
 - @ Usuário
 - @ País
 - @ Estado
 - @ Cidade
 - @ Perfil
 - @ Atividade
 - @ Categoria
 - @ Empresa
 - @ Funcionário
 - @ Produto
- Conteúdo principal com abas:
 - Empresas Cadastradas
 - Consertos em Atraso
 - Consertos a Serem Retirados
 - Consertos retirados e Pagos
- Tabela de Empresas Cadastradas:

Código	Empresa	E-Mail	Site	Data Cadastro	Situação
1	Casol	pedro.silva@casol.com.br	www.casol.com.br	01/01/2013	Ativada
4	1054	digofang@gmail.com	qwq	06/12/2013	Desativada
6	fda	digofang@gmail.com	www.havan.com.br	20/12/2013	Desativada
2	Renner	renner@renner.com.br	www.renner.com.br	01/01/2013	Ativada
3	492	havan@havan.com.br	www.havan.com.br	01/01/2013	Desativada

Figura 12. Tela principal com os menus do sistema web

O sistema disponibiliza a tela para cadastrar um ou mais usuário, com os campos matrícula, nome, login, e-mail, senha, confirme a senha e perfil, após preencher corretamente o cadastro e pressionar o botão cadastra é informado ao usuário que o cadastro foi efetivado com sucesso.

Figura 13. Tela de cadastro de usuário do sistema web

O sistema disponibiliza a tela para cadastrar um ou mais países, o código é gerado automaticamente pelo sistema e serve para identificar o registro no banco de dados, o campo nome pode ser preenchido com letras, o campo DDI aceita somente números, data cadastro e data de alteração são alterado automaticamente pelo sistema e não é permitido edita-los manualmente.

Figura 14. Tela de cadastro de país do sistema web

O sistema disponibiliza a tela para cadastrar um ou mais estados, os campos código, data cadastro e data alteração são alterado automaticamente pelo sistema e não podem ser editados manualmente, o campo nome pode ser preenchido com letras, o campo uf aceita somente duas letras não permite número, no campo país é possível selecionar o país cadastrado.

Figura 15. Tela de cadastro de estado do sistema web

O sistema disponibiliza a tela para cadastrar uma ou mais cidades, os campos código, data cadastro e data alteração são alterado automaticamente pelo

sistema e não podem ser editados manualmente, o campo nome pode ser preenchido com letras, o campo ddd aceita número, no campo estado é possível selecionar o estado cadastrado.

Cadastro de Cidade

Código: automático

Nome:

DDD:

Estado: Selecione ▼

Data Cadastro: Dec 11, 2013

Data Alteração: Dec 11, 2013

Figura 16. Tela de cadastro de cidade do sistema web

O sistema disponibiliza a tela para cadastrar um ou mais perfils, os campos código, data cadastro e data alteração são alterado automaticamente pelo sistema e não podem ser editados manualmente, o campo nome pode ser preenchido com letras.

Cadastro de Perfil

Código: automático

Nome:

Data Cadastro: 11/12/2013

Data Alteração: 11/12/2013

Figura 17. Tela de cadastro de perfil do sistema web

O sistema disponibiliza a tela para cadastrar uma ou mais atividades, os campos código, data cadastro e data alteração são alterado automaticamente pelo sistema e não podem ser editados manualmente, o campo nome pode ser preenchido com letras.

Cadastro de Atividade

Código: automático

Nome:

Data Cadastro: 11/12/2013

Data Alteração: 11/12/2013

Figura 18. Tela de cadastro de atividade do sistema web

O sistema disponibiliza a tela para cadastrar uma ou mais categorias, os campos código, data cadastro e data alteração são alterado automaticamente pelo

sistema e não podem ser editados manualmente, o campo nome pode ser preenchido com letras.



Figura 19. Tela de cadastro de categoria do sistema web

O sistema disponibiliza a tela para cadastrar uma ou mais empresas, os campos código, data cadastro e data alteração são alterado automaticamente pelo sistema e não podem ser editados manualmente, para cadastrar uma empresa os campos nome, cidade, atividade, CNPJ, IE, CEP, logradouro, bairro, número, complemento, contato, e-mail, rede social, site, localização, logo, ativo, ranking devem ser preenchidos/selecionados, o sistema não permite que duas empresas iguais sejam cadastradas, empresa que estiver inativa não será exibida no dispositivo móvel, após preencher todos os campos e pressionar o botão cadastrar, é informado ao usuário que o registro foi gravado com sucesso.

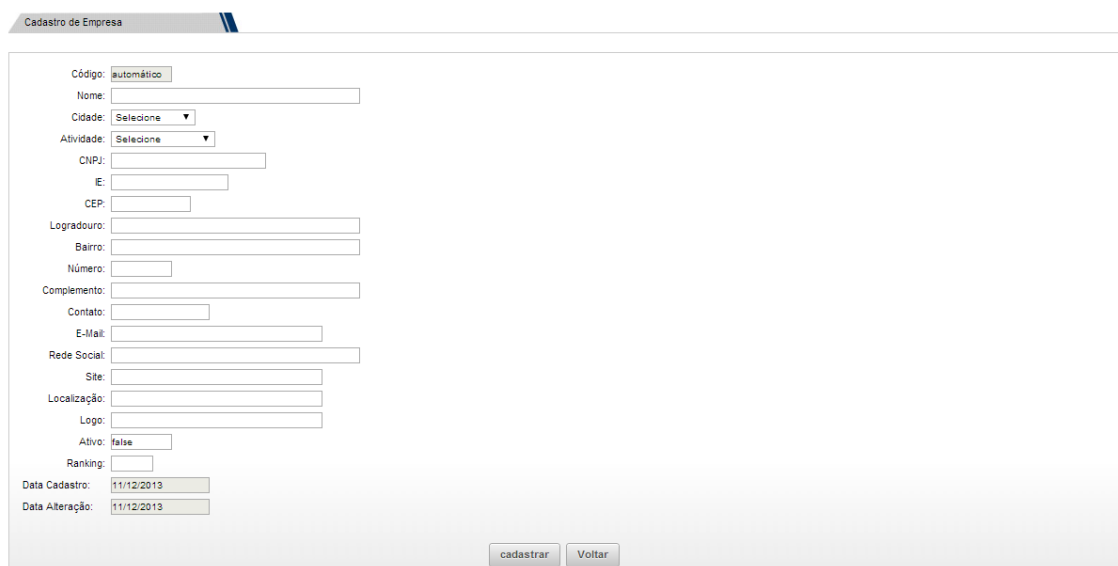


Figura 20. Tela de cadastro de empresa do sistema web

O sistema disponibiliza a tela para cadastrar um ou mais funcionários, os campos código, data cadastro e data alteração são alterado automaticamente pelo sistema e não podem ser editados manualmente, para cadastrar um funcionário os campos nome, cidade, empresa, CPF, RG, data nascimento, CEP, logradouro, bairro, número, complemento, contato, e-mail, rede social, ativo devem ser preenchidos selecionados, sistema não permite duplica registro, o funcionário cadastrado ficará responsável por cadastrar outros produtos, funcionário inativo não pode cadastrar produtos no sistema, uma empresa pode ter um ou mais funcionários.

Cadastro de Funcionário

Código: automático

Nome:

Cidade:

Empresa:

CPF:

RG:

Data Nascimento:

CEP:

Logradouro:

Bairro:

Número:

Complemento:

Contato:

E-Mail:

Rede Social:

Ativo: ☐

Data Cadastro: 11/12/2013

Data Alteração: 11/12/2013

Figura 21. Tela de cadastro de funcionário do sistema web

O sistema disponibiliza a tela para cadastrar um ou mais produtos, os campos código, data cadastro e data alteração são alterado automaticamente pelo sistema e não podem ser editados manualmente, para cadastrar um produto os campos nome, funcionário, empresa, categoria, descrição, informações técnicas, quantidade, preço normal, preço promoção data de validade, ativo, ranking, foto devem ser preenchidos/selecionados, somente será apresentado no dispositivo móvel produtos ativo, sistema não permite cadastrar produto com preço promoção maior que preço normal, após preencher corretamente todos os campos e pressionar o botão cadastrar sistema informa ao usuário que o registro foi efetivado com sucesso.

Cadastro de Produto

Código: automático

Nome:

Funcionário:

Empresa:

Categoria:

Descrição:

Inf. Técnicas:

Quantidade:

Preço Normal:

Preço Promoção:

Data Validade:

Ativo:

Ranking:

Foto:

Data Cadastro:

Data Alteração:

Figura 22. Tela de cadastro de produto do sistema web

5. CONSIDERAÇÕES FINAIS

A ideia principal do sistema foi atingida, proporcionar ao usuário um mecanismo para busca de produtos em promoção amigável e prática, documentações foram construídas para apoiar sua concepção dentre o surgimento da proposta a sua finalização.

REFERÊNCIA

SOUZA, Jorgas do Nascimento. **AS DIFERENÇAS ENTRE O DESENVOLVIMENTO TRADICIONAL E O MOBILE**. 2011. 87 f. TCC (Graduação) - Curso de Processamento de Dados, Faculdade de Tecnologia de São Paulo, São Paulo, 2011. Disponível em: <<http://www.fatecsp.br/dti/tcc/tcc0029.pdf>>. Acesso em: 02 ago. 2014.

RODRIGUES, Guilherme Rodrigues e. **SMARTPHONES E SUAS TECNOLOGIAS**. 2009. 77 f. TCC (Graduação) - Curso de Engenharia Elétrica Com ênfase em Eletrônica, Universidade de São Paulo, São Carlos, 2009. Disponível em: <<http://www.tcc.sc.usp.br/tce/disponiveis/18/180450/tce-23042010-094332/?&lang=br>>. Acesso em: 02 ago. 2014.

GONÇALVES, Julio Cesar. **USO DA PLATAFORMA ANDROID EM UM PROTÓTIPO DE APLICATIVO COLETOR DE CONSUMO DE GÁS NATURAL**. 2011. 63 f. TCC (Graduação) - Curso de Especialização em Tecnologia Java, Universidade Tecnológica Federal do Paraná, Curitiba, 2011. Disponível em: <http://www2.dainf.ct.utfpr.edu.br/esp/monografias-de-especializacao-da-turma-vi-2010-2011/CT_JAVA_VI_2010_10.PDF/view>. Acesso em: 02 ago. 2014.

MARTINS, Rafael J. Werneck de A. **Desenvolvimento de Aplicativo para Smartphone com a Plataforma Android**. 2009. 50 f. TCC (Graduação) - Curso de Engenharia de Computação, Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, 2009. Disponível em: <<http://www.icad.puc-rio.br/~projetos/android/files/monografia.pdf>>. Acesso em: 02 ago. 2014.

NOGUEIRA, Rodolpho Bruno dos Santos. **DESENVOLVIMENTO DE UM SISTEMA DE APOIO A EVENTOS PARA UMA PLATAFORMA ANDROID**. 2012. 164 f. TCC (Graduação) - Curso de Bacharelado em Ciência da Computação, Centro Universitário Eurípides de Marília, Marília, 2012. Disponível em: <<http://hdl.handle.net/11077/873>>. Acesso em: 04 ago. 2014.

ALINE JESUS (Brasil). Compre nas Casas Bahia diretamente de seu smartphone ou tablet. 2014. Disponível em: <<http://www.techtudo.com.br/tudo-sobre/s/casas-bahia.html>>. Acesso em: 06 ago. 2014.

B2W DIGITAL (Brasil). Submarino. 2014. Disponível em: <<https://play.google.com/store/apps/details?id=com.b2w.submarino>>. Acesso em: 06 ago. 2014.

LOJAS RENNEN (Brasil). Lojas Renner. 2014. Disponível em: <https://play.google.com/store/apps/details?id=br.com.lojasrenner&hl=pt_BR>. Acesso em: 06 ago. 2014.

SILVA, Renato Alves da. **MARKETING: A PROPAGANDA É A ALMA DO NEGÓCIO. A IMPORTÂNCIA DESTA FERRAMENTA DE MARKETING NAS ORGANIZAÇÕES.** 2011. 59 f. TCC (Graduação) - Curso de Administração, Faculdade Cenecista de Capivari, Capivari, 2011. Disponível em: <http://www.cneccapivari.br/libdig/index.php?option=com_rubberdoc&view=doc&id=425&format=raw>. Acesso em: 08 ago. 2014.

TAMARE, RODRIGO YOSHIO & LIMA, PABLO RODRIGUES. Web Services: uma nova visão da arquitetura de aplicações distribuídas na Internet. Revista Científica Eletrônica de Sistemas de Informação, 2005.

INSTITUTE, Project Management. Um Guia do Conhecimento em Gerenciamento de Projetos (Guia PMBOK). 4. ed. Newtown Square, Pennsylvania: Saraiva, 2008. 459 p. Disponível em: <[http://www.tecgraf.puc-rio.br/~bia/Estudo/PMBOK 2008 Português.pdf](http://www.tecgraf.puc-rio.br/~bia/Estudo/PMBOK%202008%20Portugu%C3%AAs.pdf)>. Acesso em: 16 ago. 2014.

APÊNDICE**APÊNDICE A - 1 - Exemplo Visão**

Visão
Versão 2.1

Histórico da Revisão

Data	Versão	Descrição	Autor
10/08/2013	1.0	Versão inicial	Maicon Alexander Illes Fang
11/12/2013	2.0	Versão secundaria	Maicon Alexander Illes Fang
19/02/2014	2.1	Remoção de algumas atividades	Maicon Alexander Illes Fang

Índice Analítico

1.	Introdução	38
1.1	Referências	38
2.	Posicionamento	38
2.1	Descrição do Problema	38
2.2	Sentença de Posição do Produto	39
3.	Descrições dos Envolvidos e Usuários	39
3.1	Resumo dos Envolvidos	39
3.2	Resumo dos Usuários	39
3.3	Ambiente do Usuário	40
3.4	Resumo das Principais Necessidades dos Envolvidos ou Usuários	40
3.5	Alternativas e Concorrência	41
4.	Visão Geral do Produto	41
4.1	Perspectiva do Produto	43
4.2	Suposições e Dependências	43

Visão

1. Introdução

A finalidade deste documento é coletar, analisar e definir as necessidades e características de nível superior do mBestProducts. Este documento relaciona os recursos de que os usuários e envolvidos no problema necessitam para melhorar a forma de busca de informações. Os detalhes de como o mBestProducts atende a essas demandas estão descritos nas especificações suplementares e de caso de uso.

1.1 Referências

Nenhuma.

2. Posicionamento**2.1 Descrição do Problema**

O problema	Falta de simplicidade na apresentação de produtos em promoção em dispositivos móveis.
Afeta	Usuários comuns.
cujo impacto é	Não ter um sistema que exibe de forma simples e fácil promoções dos produtos em dispositivos moveis.
uma boa solução seria	Oferecer um sistema que exibisse os produtos em promoção de forma fácil e simples.

2.2 Sentença de Posição do Produto

Para	Usuários
Quem	Coordenadores, colaboradores pertencentes a uma mesma empresa.
O (nome do produto)	É um software.
Que	Exibe produtos que estão em promoção.
Diferente de	Exibir através de sites na internet.
Nosso produto	Um ambiente amigável e confiável.

3. Descrições dos Envolvidos e Usuários

O sistema está direcionado a todos os usuários que possuam um dispositivo móvel com sistema operacional android e com a aplicação instalada no aparelho.

3.1 Resumo dos Envolvidos

Nome	Descrição	Responsabilidades
Usuário	Acessa a aplicação.	Realiza busca de produtos em promoção.
Administrador	Cadastra, altera, lista e inativa e exclui.	Cadastra, altera, lista e inativa produtos e empresa no sistema.

3.2 Resumo dos Usuários

Nome	Descrição	Envolvido
------	-----------	-----------

Usuário	Acessa à aplicação.	Auto representado
Administrador	Cadastra, altera, lista e inativa empresa e produtos no sistema.	Auto representado

3.3 Ambiente do Usuário

Usuários terão acesso a um aplicativo onde será disponibilizado as empresas e produtos, poderão acessar de qualquer lugar desde que tenha acesso a internet.

Administrador terá acesso a um web-site onde terá um login e senha, será possível criar, alterar, listar e inativar uma empresa, categoria, também poderá: cadastrar, alterar, listar, excluir e inativar produtos no sistema.

3.4 Resumo das Principais Necessidades dos Envolvidos ou Usuários

Necessidade	Prioridade	Preocupações	Solução Atual	Soluções Propostas
Cadastrar Pais.	Alta	Identificação dos usuários envolvidos	Possui Cadastro	Criar página onde os administradores cadastrem o país.
Cadastrar Estado.	Alta	Identificação dos usuários envolvidos	Possui Cadastro	Criar página onde os administradores cadastrem o estado.
Cadastrar Cidade.	Alta	Identificação dos usuários envolvidos	Possui Cadastro	Criar página onde os administradores cadastrem a cidade.
Cadastrar Empresa.	Alta	Identificação dos usuários envolvidos	Possui Cadastro	Criar página onde os administradores cadastrem a empresa.
Cadastrar Funcionário.	Alta	Identificação	Possui Cadastro	Criar página onde os

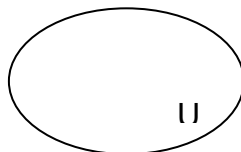
		usuários dos envolvidos		administradores cadastram o funcionário.
Cadastrar Usuário.	Alta	Identificação dos usuários dos envolvidos	Possui Cadastro	Criar página onde os administradores cadastram o usuário.
Cadastrar Perfil.	Alta	Identificação dos usuários dos envolvidos	Possui Cadastro	Criar página onde os administradores cadastram o perfil.
Cadastrar Atividade.	Alta	Identificação dos usuários dos envolvidos	Possui Cadastro	Criar página onde os administradores cadastram a atividade.
Cadastrar Categoria.	Alta	Identificação dos usuários dos envolvidos	Possui Cadastro	Criar página onde os administradores cadastram a categoria.
Cadastrar Produto.	Alta	Identificação dos usuários dos envolvidos	Possui Cadastro	Criar página onde os administradores cadastram o produto.

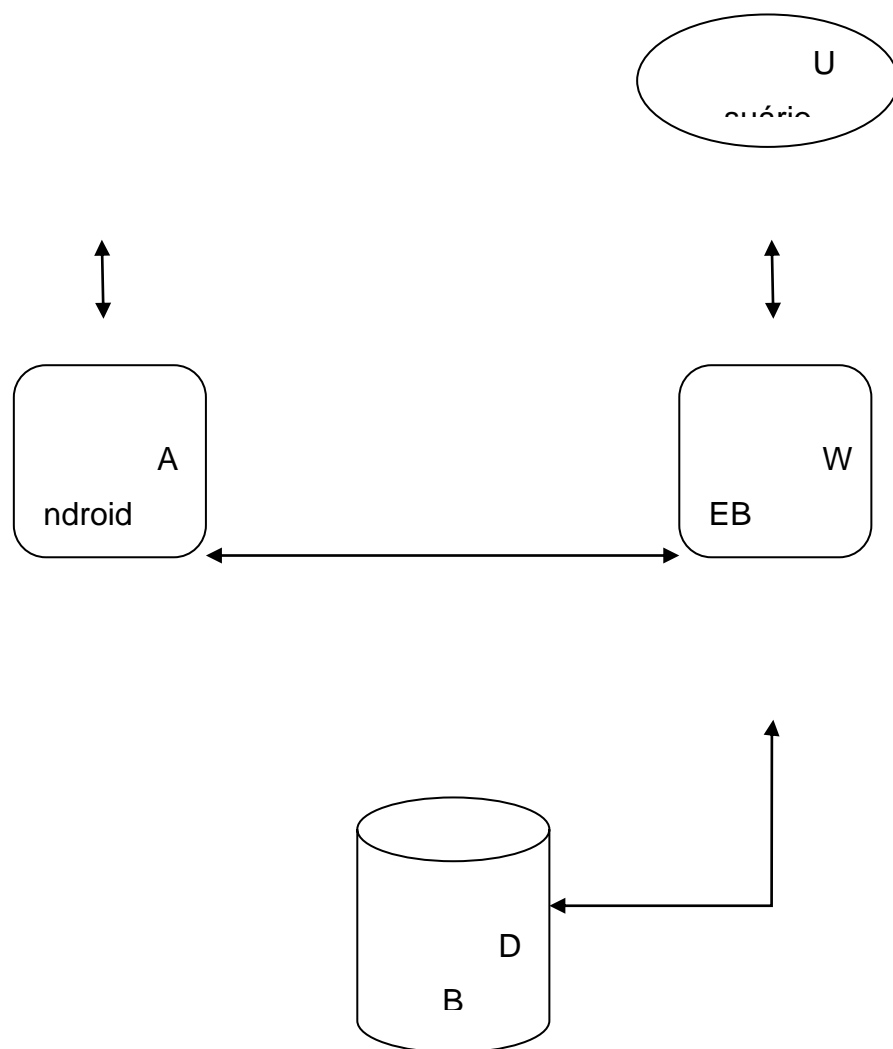
3.5 Alternativas e Concorrência

Atualmente não existem ferramentas que atendam a esta demanda em específico.

4. Visão Geral do Produto

O sistema terá páginas web onde os administradores terão um local apropriado e restrito para interação e compartilhamento de informações. Graficamente o sistema pode ser representado da seguinte forma:





4.1 Perspectiva do Produto

Benefícios para o usuário	Recursos do sistema
Agilidade no retorno dos produtos em promoção.	Aplicativos android que lista os produtos.
Cadastros das empresas no site.	Páginas para cadastro de empresas.
Cadastros dos funcionários no site.	Páginas para cadastro de funcionários.
Cadastros dos produtos no site.	Páginas para cadastro de produtos.

4.2 Suposições e Dependências

O sistema precisará de um servidor Web disponível 24 horas por dia, base de dados, Java, também será necessário um domínio registrado. Administradores precisam estar cadastrados com um nome de usuário padrão e senha para que possam ter acesso via internet ao sistema. Uma vez o sistema implantado, o sistema deve ser compatível com o padrão Web existente (HTML, Java, TCP/IP), acessado com o navegador Google Chrome. O sistema precisará também de um aplicativo que requisitará pedidos de informações a base central do servidor web (web services), o aplicativo será feito em linguagem Java, será instalado em dispositivos moveis com sistema operacional Android versão 4.0 acima.

APÊNDICE B - 2 - Template Regras de Negócio

mBestProducts
Regras de Negócios

Versão 2.1

Histórico da Revisão

Data	Versão	Descrição	Autor
11/08/2013	1.0	Primeira revisão	Maicon Fang
16/08/2013	1.1	Segunda revisão	Maicon Fang
11/08/2013	2.0	Terceira revisão	Maicon Fang
19/08/2014	2.1	Ajustes de atividades	Maicon Alexander Illes Fang

Regras de Negócios

Introdução

Regras de Negócios oferece uma visão geral de todo o documento, contempla as principais regras que o sistema precisa ter para ser desenvolvido.

Finalidade

Descrever as regras de negócio referente ao sistema mBestProducts como um todo. Sendo dividido em duas partes. WEB que é responsável por alimentar as informações na base de dados, e o aplicativo feito em Java Android que irá buscar as informações na base de dados.

Definições Sistema Android

Neste tópico serão apresentados as regras de negócios separadas por dois grupos o primeiro Sistema Android e o segundo Sistema WEB

Sistema Android

R4 – Data da validade da promoção do produto vencida

O sistema não deve apresentar o produto na tela do celular android.

R1 - Não possuir empresa cadastrada

O sistema deve exibir a mensagem “Não possui empresa cadastrada”

R3 – Não possuir produto cadastrado

O sistema deve exibir a mensagem “Não possui produto cadastrado”

R9 – Sem conexão com a Internet

O sistema deverá aprestar a mensagem “Sem conexão com a internet”

R10 – Empresa não ativada

O sistema não exibe a/s empresa/s na aplicação Android.

R11 – Produto não ativado

O sistema não exibe o/s produto/s na aplicação Android.

Sistema WEB

R6 – Tamanho da logo da empresa

O sistema deve permitir cadastrar a logo da empresa com o tamanho máximo de 100 kbytes.

R7 – Formato da logo da empresa

O sistema deve permitir cadastrar a logo da empresa com o formato JPEG.

R11 – CNPJ duplicados

O sistema não deve permitir CNPJ duplicados e exibir a mensagem “CNPJ já cadastrado”

R13 - Nome de Armazenamento de imagens dos produtos

As imagens dos produtos serão armazenadas no servidor de internet, iniciando sempre com “cnpj_produto_nome”.

R14 - Armazenamento de imagens das empresas

As imagens dos produtos serão armazenadas no servidor de internet, iniciando sempre com “cnpj_empresa_nome”.

R16 - Tamanho da imagem do produto deve possuir 100 kbytes.

O sistema deve permitir cadastrar a imagem do produto com o tamanho máximo de 100 kbytes.

R17 – Campo Data Cadastro

Ao efetivar um cadastro a data de cadastro não é mais alterado

R19 – Campo Preço Promoção

O sistema não permite que valor da promoção seja maior que o preço normal do produto

R20 – Campo Data Validade

O sistema não permite que a Data de Validade seja menor que a data corrente/atual.

R21 – CPF duplicados

O sistema não permite CPF duplicados e exibe a mensagem “CPJ já cadastrado”.

R22 – Validação de CPF

O sistema exibe a mensagem “CPF inválido”

R23 – Validação de CNPJ

O sistema exibe a mensagem “CNPJ inválido”

R24 – Idade de nascimento do funcionário

Ao cadastrar um funcionário menor de 18 anos, o sistema exibe a mensagem “Não é possível cadastrar um Funcionário menor de 18 anos de idade”.

APÊNDICE C - 3 - Template Glossario

Glossário de Negócios

Versão 2.1

Histórico da Revisão

Data	Versão	Descrição	Autor
12/08/2013	1.0	Primeira revisão	Maicon Fang
16/08/2013	1.1	Segunda revisão (Aval Professor)	Maicon Fang
11/12/2013	2.0	Terceira revisão	Maicon Fang
19/02/2014	2.1	Ajustes de itens	Maicon Alexander Illes Fang

Introdução

Oferecer uma visão geral de todo o documento. Apresenta a definição dos termos de negócio.

Finalidade

Este documento tem por finalidade definir a terminologia específica do domínio de problema, explicando termos que podem não ser familiares para o leitor das descrições de casos de uso ou de outros documentos do projeto. Também usado como um

dicionário de dados informal, capturando definições de dados para que as descrições de casos de uso e outros documentos do projeto possam se concentrar no que o sistema deve fazer com as informações.

Definições

Termos definidos aqui formam a parte essencial do documento.

ADT

ADT - Android Development Tools - (Ferramentas de desenvolvimento Android).

IDE

IDE – Integrated Development Environment - (Ambiente Integrado de Desenvolvimento).

WEB

WEB (World Wide Web - Teia Mundial)

WEB Service

Web Service é uma solução utilizada na integração de sistemas e na comunicação entre aplicações diferentes. Com esta tecnologia é possível que novas aplicações possam interagir com aquelas que já existem e que sistemas desenvolvidos em plataformas diferentes sejam compatíveis.

JSF

JSF (Java Server Faces)

Java

Java é uma linguagem de programação orientada a objeto.

Internet

A Internet é o maior conglomerado de redes de comunicações em escala mundial, vários computadores e dispositivos conectados em uma rede mundial.

Android

Android é um sistema operacional baseado no núcleo do Linux para dispositivos móveis, desenvolvido pela Open Handset Alliance, liderada pelo Google e outras empresas.

APÊNDICE D - 4 - Template Casos de Uso Negociais

Versão 2.1

Cliente: Lojas em geral

Processos a serem contratados com a empresa de desenvolvimento de software

EMPRESA

Cadastro completo da empresa.

FUNCIONARIO

Cadastro completo do funcionário.

PRODUTO

Cadastro completo do produto.

PAIS

Cadastro completo do país.

ESTADO

Cadastro completo do estado.

CIDADE

Cadastro completo da cidade.

CATEGORIA

Cadastro completo da categoria.

PERFIL

Cadastro completo do perfil.

USUARIO

Cadastro completo do usuário.

ATIVIDADE

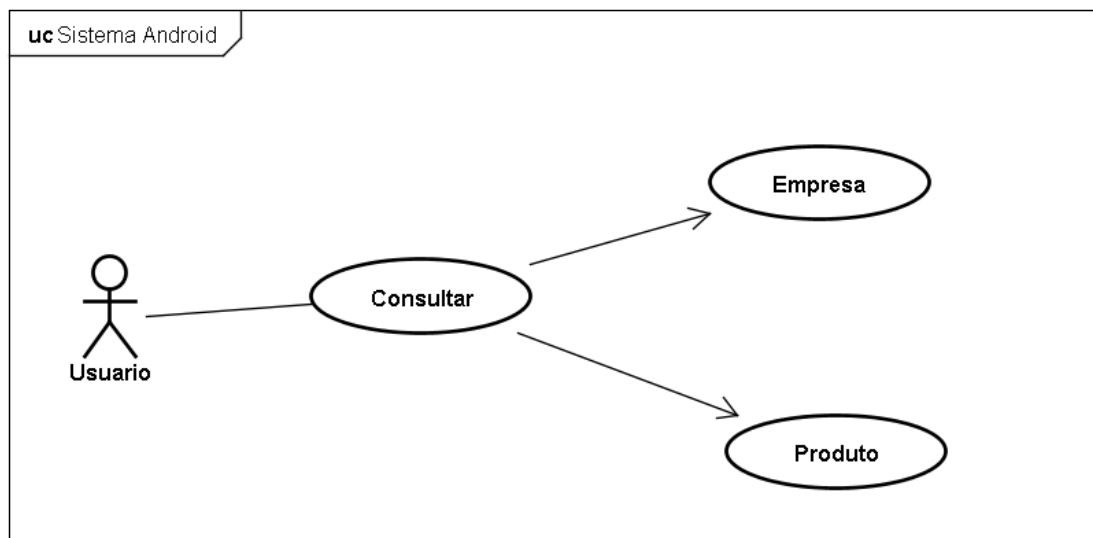
Cadastro completo da atividade da empresa.

EXIBICAO DOS DADOS DAS EMPRESAS NA TELA DO CELULAR

Empresas cadastradas exibidas na tela do celular.

EXIBICAO DOS DADOS DOS PRODUTOS NA TELA DO CELULAR

Produtos ativos e com data de validade da promoção igual ou superior a data corrente/atual exibidos na tela do celular.



UC500 – Exibir empresa/s na aplicação Android.

Controle do Documento

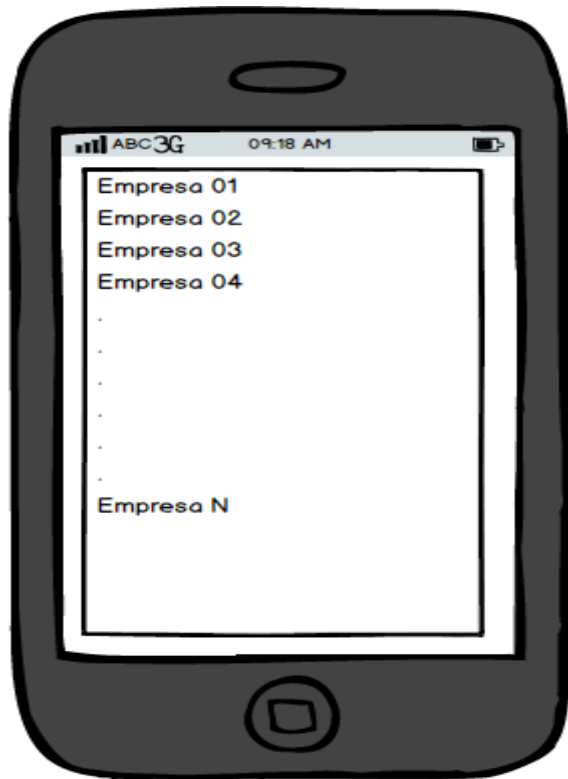
Versão	Autor	Data	Descrição
1.0	Maicon Fang	12/08/2013	Elaboração
1.1	Maicon Fang	16/08/2013	Correções
2.0	Maicon Fang	12/12/2013	Correções
2.1	Maicon Alexander Illes Fang	19/02/2014	Remoção do caso de uso 502

Descrição

Este caso de uso serve para exibir a/s empresa/s na aplicação Android.

Data Views

DV1 - Tela de exibição da empresa.



Pré-condições

Este caso de uso pode iniciar somente se:

Existir conexão com a internet.

A aplicação estiver previamente instalada no dispositivo móvel.

Ter acessado a aplicação.

Web Service estiver ativo.

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

Ter exibido a/s empresa/s.

Ator Primário

Usuário

Fluxo de Eventos Principal

Ao inicializar o aplicativo as empresas são carregadas (DV1), (R10).

O usuário seleciona a empresa (A1), (A2).

O sistema chama o Use Case 501

O caso de uso é finalizado.

Fluxos Alternativos

A1: Botão “Voltar” pressionado.

O sistema retorna para tela anterior.

O caso de uso é finalizado

A2: Botão “Home” pressionado.

O sistema retorna para tela principal do dispositivo móvel.

O caso de uso é finalizado

Fluxos de Exceção

Não aplicável.

UC501 – Exibir produto/s na aplicação Android.

Controle do Documento

Versão	Autor	Data	Descrição
1.0	Maicon Fang	12/08/2013	Elaboração
1.1	Maicon Fang	16/08/2013	Correções

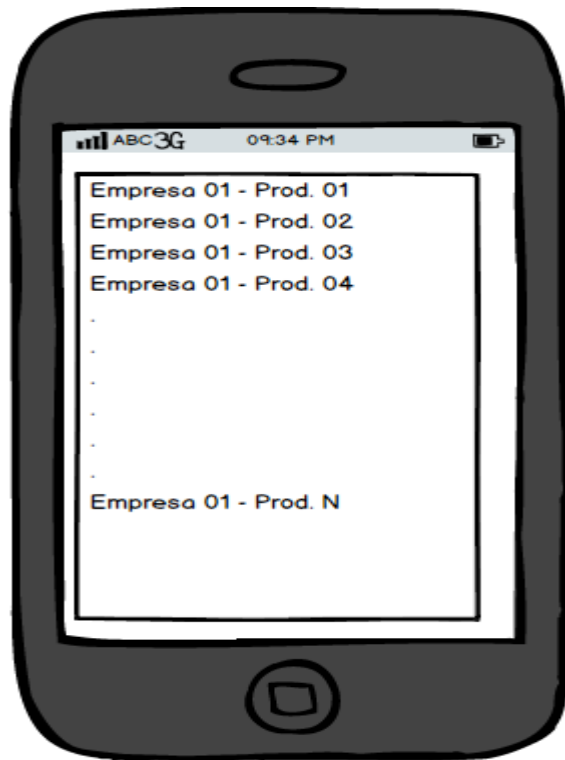
2.0	Maicon Fang	12/12/2013	Correções
2.1	Maicon Alexander Illes Fang	19/02/2014	Correções

Descrição

Este caso de uso serve para exibir os produtos na aplicação Android.

Data Views

DV2 - Tela de exibição dos produtos.



Pré-condições

Este caso de uso pode iniciar somente se:

A aplicação estiver previamente instalada no dispositivo móvel.

Ter acessado á aplicação.

Ter executado o Use Case 500.

Web Service estiver ativo

Pós-condições

Após o fim normal deste caso de uso o sistema deve:
Ter exibido os produtos.

Ator Primário

Usuário

Fluxo de Eventos Principal

Ao clicar na empresa o/s produto/s são exibidos (DV1), (R4), (R11), (E1), (E2)
O usuário seleciona o produto.
O sistema chama o Use case 502
O caso de uso é finalizado.

Fluxos Alternativos

A1: Botão “Voltar” pressionado.
O sistema retorna para tela anterior.
O sistema apresenta a tela com as empresas cadastradas.
O caso de uso é finalizado.

A2: Botão “Home” pressionado.

O sistema retorna para tela principal do dispositivo móvel.

O caso de uso é finalizado.

Fluxos de Exceção

Não aplicável.

UC502 – Exibir um produto na aplicação Android.

Controle do Documento

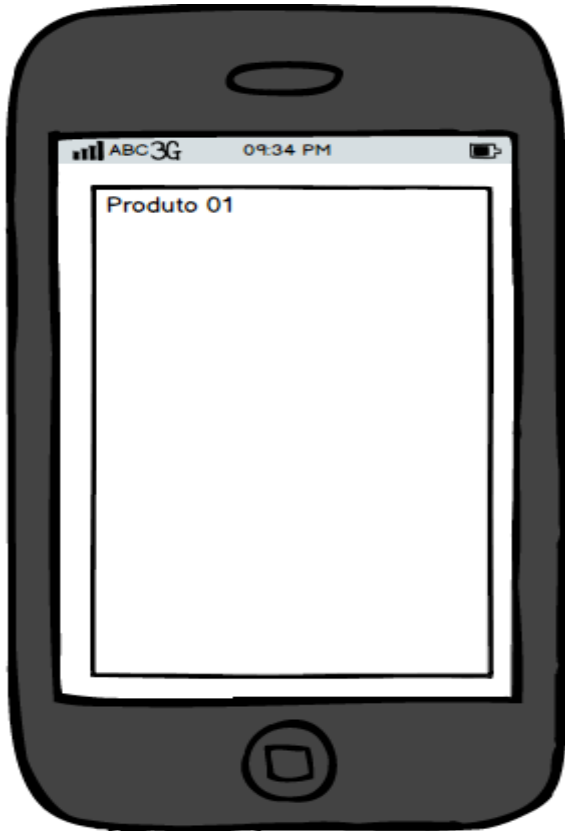
Versão	Autor	Data	Descrição
1.0	Maicon Fang	12/08/2013	Elaboração
1.1	Maicon Fang	16/08/2013	Correções
2.0	Maicon Fang	12/12/2013	Correções
2.1	Maicon Alexander Illes Fang	19/02/2014	Correções

Descrição

Este caso de uso serve para exibir um único produto na aplicação Android.

Data Views

DV1 - Tela de exibição do produto.



Pré-condições

Este caso de uso pode iniciar somente se:

Existir conexão com a internet.

A aplicação estiver previamente instalada no dispositivo móvel.

Ter acessado a aplicação.

Ter executado o Use Case 501.

O Web Service estiver ativo.

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

Ter exibido o produto.

Ator Primário

Usuário

Fluxo de Eventos Principal

À aplicação exibe o produto selecionado (DV1), (E1), (E2)

O caso de uso é finalizado

Fluxos Alternativos

A1: Botão “Voltar” pressionado.

O sistema retorna para tela anterior.

O caso de uso é finalizado

A2: Botão “Home” pressionado.

O sistema retorna para tela principal do dispositivo móvel.

O caso de uso é finalizado

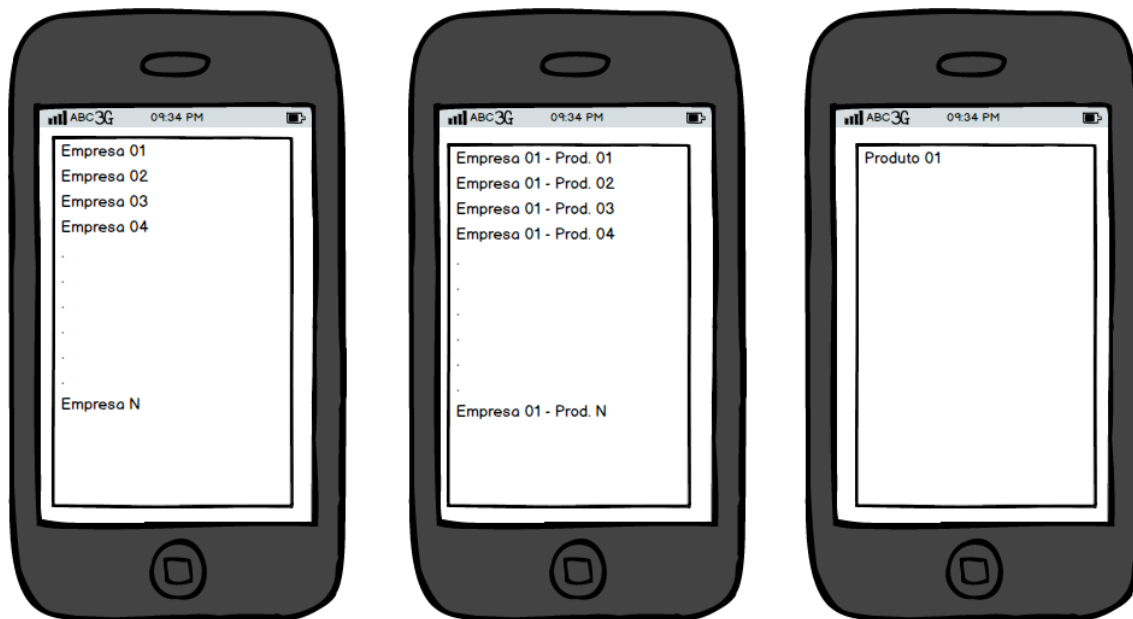
Fluxos de Exceção

Não aplicável.

APÊNDICE E - 5 - Template Protótipo de Interfaces


Versão 2.1

Telas – Android.



Tela – Apresentação dos produtos da empresa.

Telas – WEB.


 ACESSO RESTRITO


Login:

Senha:

[Esqueci minha senha.](#)

Tela – Login



Sair 

Usuário Logado: Maicon Fang

Home Produtos Empresa Funcionários

Menu

- Usuário
- Pais
- Estado
- Cidade
- Perfil
- Atividade
- Categoria
- Empresa
- Funcionário
- Produto

Empresas Cadastradas Consertos em Atraso Consertos a Serem Retirados Consertos retirados e Pagos

Código	Empresa	E-Mail	Site	Data Cadastro	Situação
1	Casol	pedro.silva@casol.com.br	www.casol.com.br	01/01/2013	Ativada
4	1054	digofang@gmail.com	qwq	06/12/2013	Desativada
6	fda	digofang@gmail.com	www.havan.com.br	20/12/2013	Desativada
2	Renner	renner@renner.com.br	www.renner.com.br	01/01/2013	Ativada
3	492	havan@havan.com.br	www.havan.com.br	01/01/2013	Desativada

Tela – Home

Cadastro de Usuário

Matricula:

Nome:

Login:

E-Mail:

Senha:

Confirme a Senha:

Perfil:

Tela – Cadastro - Usuários.

Cadastro de Pais

Código: automático

Nome:

DDI:

Data Cadastro: Dec 11, 2013

Data Alteração: Dec 11, 2013

Tela – Cadastro – Pais

Cadastro de Estado

Código: automático

Nome:

UF:

País:

Data Cadastro: 11/12/2013

Data Alteração: 11/12/2013

Tela – Cadastro – Estado.

Cadastro de Cidade

Código: automático

Nome:

DDD:

Estado:

Data Cadastro: Dec 11, 2013

Data Alteração: Dec 11, 2013

Tela – Cadastro – Cidade.

Cadastro de Perfil

Código: automático

Nome:

Data Cadastro: 11/12/2013

Data Alteração: 11/12/2013

Tela – Cadastro – Perfil.

Cadastro de Atividade

Código: automático

Nome:

Data Cadastro: 11/12/2013

Data Alteração: 11/12/2013

Tela - Cadastro – Atividade.

Cadastro de Categoria

Código: automático

Nome:

Data Cadastro: 11/12/2013

Data Alteração: 11/12/2013

Tela – Cadastro – Categoria.

Cadastro de Empresa

Código: automático

Nome:

Cidade:

Atividade:

CNPJ:

IE:

CEP:

Logradouro:

Bairro:

Número:

Complemento:

Contato:

E-Mail:

Rede Social:

Site:

Localização:

Logo:

Ativo:

Ranking:

Data Cadastro: 11/12/2013

Data Alteração: 11/12/2013

Tela – Cadastro – Empresa.

Cadastro de Funcionário

Código: automático

Nome:

Cidade:

Empresa:

CPF:

RG:

Data Nascimento:

CEP:

Logradouro:

Bairro:

Número:

Complemento:

Contato:

E-Mail:

Rede Social:

Ativo:

Data Cadastro: 11/12/2013

Data Alteração: 11/12/2013

Tela – Cadastro – Funcionário.

Cadastro de Produto

Código: automático

Nome:

Funcionário:

Empresa:

Categoria:

Descrição:

Inf. Técnicas:

Quantidade:

Preço Normal:

Preço Promoção:

Data Validade:

Ativo:

Ranking:

Foto:

Data Cadastro:

Data Alteração:

Tela – Cadastro – Produto.

APÊNDICE F - 6 - Template Especificação de Caso de Uso

Especificação de Caso de Uso

Versão 2.1

Histórico da Revisão

Data	Versão	Descrição	Autor
10/08/2013	1.0	Versão inicial	Maicon Alexander Illes Fang
11/12/2013	2.0	Versão secundaria	Maicon Alexander Illes Fang
19/02/2014	2.1	Ajustes e inserção de novos casos de usos.	Maicon Alexander Illes Fang

UC001 – Manter Empresa

Controle do Documento

Versão	Autor	Data	Descrição
1.0	Maicon Fang	12/08/2013	Elaboração
1.1	Maicon Fang	16/08/2013	Correções
2.0	Maicon Fang	13/12/2013	Correções

Descrição

Este caso de uso serve para cadastrar uma empresa.

Data Views

DV1 – Tela cadastro da empresa.

Cadastro de Empresa

Código:

Nome:

Cidade:

Atividade:

CNPJ:

IE:

CEP:

Logradouro:

Bairro:

Número:

Complemento:

Contato:

E-Mail:

Rede Social:

Site:

Localização:

Logo:

Ativo:

Ranking:

Data Cadastro:

Data Alteração:

Pré-condições

Este caso de uso pode iniciar somente se:

O sistema tiver executado o UC011 - Login

O usuário possuir permissão para cadastrar uma empresa.

O usuário tiver cadastrado uma cidade e atividade.

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

Ter salvo os dados da empresa.

Ator Primário

Administrador

Fluxo de Eventos Principal

- O sistema apresenta a tela (DV1)
- O sistema habilita o botão Cadastrar.
- O sistema carrega a data corrente no campo Data Criação (R17).
- O sistema carrega a data corrente no campo Data Alteração (E1).
- O sistema desabilita para edição o campo Data Criação.
- O sistema desabilita para edição o campo Data Alteração.
- O sistema desabilita para edição o campo Código.
- O sistema carrega a lista de Cidades cadastradas.
- O sistema carrega a lista de Atividades cadastradas.
- O usuário preenche o campo Nome.
- O usuário seleciona o campo Cidade.
- O usuário seleciona o campo Atividade.
- O usuário seleciona o campo Foto.
- O usuário preenche o campo CNPJ (R11).
- O usuário preenche o campo IE.
- O usuário preenche o campo CEP.
- O usuário preenche o campo Logradouro.
- O usuário preenche o campo Número.
- O usuário preenche o campo Complemento.

- O usuário preenche o campo Bairro.
- O usuário preenche o campo Telefone.
- O usuário preenche o campo Contato.
- O usuário preenche o campo Site.
- O usuário pressiona o botão Cadastrar. (A3)
- O caso de uso é finalizado.

Fluxos Alternativos

- A3: Botão Voltar pressionado.
- O usuário pressiona o botão voltar.
- O sistema retorna a tela anterior.

Fluxos de Exceção

- E1. Campo Data Alteração.
- Ao editar um registro o campo data alteração recebe a data corrente/atual.

UC002 – Manter Funcionário

Controle do Documento

Versão	Autor	Data	Descrição
1.0	Maicon Fang	12/08/2013	Elaboração
1.1	Maicon Fang	16/08/2013	Correções

2.0	Maicon Fang	13/12/2013	Correções
-----	-------------	------------	-----------

Especificação de Caso de Uso

Descrição

Este caso de uso serve para cadastrar um funcionário.

Data Views

DV1 – Tela cadastro do funcionário.

Cadastro de Funcionário

Código: automático

Nome:

Cidade:

Empresa:

CPF:

RG:

Data Nascimento:

CEP:

Logradouro:

Bairro:

Número:

Complemento:

Contato:

E-Mail:

Rede Social:

Ativo: ☐

Data Cadastro:

Data Alteração:

Pré-condições

Este caso de uso pode iniciar somente se:

- O sistema tiver executado o UC011 - Login
- O usuário possuir permissão para cadastrar um funcionário.
- O usuário tiver cadastrado uma cidade e empresa.

Pós-condições

- Após o fim normal deste caso de uso o sistema deve:
- Ter salvo os dados do Funcionário.

Ator Primário

Administrador

Fluxo de Eventos Principal

- O sistema apresenta a tela (DV1)
- O sistema habilita o botão Cadastrar.
- O sistema habilita o botão Voltar.
- O sistema desabilita para edição o campo Data Criação (R17).
- O sistema desabilita para edição o campo Data Alteração (E).
- O sistema desabilita para edição o campo Código.
- O sistema carrega a data corrente no campo Data Criação (A2).
- O sistema carrega a data corrente no campo Data Alteração.
- O sistema carrega uma lista de Empresas no combo Box Empresa.
- O sistema carrega uma lista de Cidades no combo Box Cidade.
- O usuário preenche o campo Nome.

- O usuário seleciona o campo Cidade.
- O usuário seleciona o campo Empresa.
- O usuário seleciona o campo Foto.
- O usuário preenche o campo CPF (R21).
- O usuário preenche o campo RG.
- O usuário preenche o campo Data Nascimento.
- O usuário preenche o campo CEP.
- O usuário preenche o campo Logradouro.
- O usuário preenche o campo Bairro.
- O usuário preenche o campo Número.
- O usuário preenche o campo Complemento.
- O usuário preenche o campo Contato.
- O usuário preenche o campo E-mail.
- O usuário preenche o campo Rede Social.
- O usuário preenche o campo Ativo.
- O usuário pressiona o botão Cadastrar. (A1)
- O caso de uso é finalizado.

Fluxos Alternativos

- A1: Botão Voltar pressionado.
- O usuário pressiona o botão voltar.
- O sistema retorna a tela anterior.

Fluxos de Exceção

E1. Campo Data Alteração.

Ao editar um registro o campo data alteração recebe a data corrente/atual.

UC003 – Manter Produto

Controle do Documento

Versão	Autor	Data	Descrição
1.0	Maicon Fang	12/08/2013	Elaboração
1.1	Maicon Fang	16/08/2013	Correções
2.0	Maicon Fang	13/12/2013	Correções

Descrição

Este caso de uso serve para cadastrar um produto

Data Views

DV1 – Tela cadastro do produto.

Cadastro de Produto

Código: automático

Nome:

Funcionário:

Empresa:

Categoria:

Descrição:

Inf. Técnicas:

Quantidade:

Preço Normal:

Preço Promoção:

Data Validade:

Ativo:

Ranking:

Foto:

Data Cadastro:

Data Alteração:

Pré-condições

Este caso de uso pode iniciar somente se:

O sistema tiver executado o UC011 - Login

O usuário possuir permissão para cadastrar um Produto.

O sistema deve ter um Funcionário, Empresa e Categoria cadastrada.

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

Ter salvo os dados do Produto.

Ator Primário

Funcionário.

Fluxo de Eventos Principal

- O sistema apresenta a tela (DV1)
- O sistema habilita o botão Cadastrar.
- O sistema habilita o botão Voltar.
- O sistema desabilita para edição o campo Data Criação.
- O sistema desabilita para edição o campo Data Alteração.
- O sistema desabilita para edição o campo Código.
- O sistema carrega a data corrente no campo Data Criação (R17).
- O sistema carrega a data corrente no campo Data Alteração (E3).
- O sistema carrega o campo ativo preenchido e desabilitado (E2).
- O sistema carrega o combo Box Categoria.
- O sistema carrega o combo Box Empresa.
- O sistema carrega o combo Box Funcionário.
- O usuário seleciona o campo Foto do Produto.
- O usuário seleciona o campo Categoria do Produto.
- O usuário preenche o campo Nome.
- O usuário preenche o campo Descrição.
- O usuário preenche o campo Especificações Técnicas.
- O usuário preenche o campo Quantidade.
- O usuário preenche o campo Preço Normal.
- O usuário preenche o campo Preço Promoção (R19).

O usuário preenche o campo Data Validade (R20).

O usuário pressiona o botão Salvar. (A1).

O caso de uso é finalizado.

Fluxos Alternativos

A1: Botão Voltar pressionado.

O usuário pressiona o botão voltar.

O sistema retorna a tela anterior.

Fluxos de Exceção

E1. Lista sem produto cadastrada

O sistema exibe a lista sem produtos cadastrados

O sistema desabilita os botões Editar, Excluir

E2. Campo ativo editar registro

O sistema habilita o campo com as opções ativo e inativo

E3. Campo Data Alteração

Ao editar um registro o campo data alteração recebe a data corrente/atual.

UC004 – Manter Categoria

Controle do Documento

Versão	Autor	Data	Descrição
1.0	Maicon Fang	12/08/2013	Elaboração
1.1	Maicon Fang	16/08/2013	Correções
2.0	Maicon Fang	13/12/2013	Correções

Descrição

Este caso de uso serve para fazer o cadastramento de uma Categoria

Data View

DV1 - Tela de cadastramento de Categoria



Cadastro de Categoria

Código: automático

Nome:

Data Cadastro: 11/12/2013

Data Alteração: 11/12/2013

Pré-condições

Este caso de uso pode iniciar somente se:

O sistema tiver executado o UC011 - Login

O usuário possuir permissão para cadastrar uma Categoria.

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

Ter salvo os dados da Categoria.

Ator Primário

Administrador

Fluxo de Eventos Principal

O sistema apresenta a tela (DV1)

O sistema habilita o botão Cadastrar.

O sistema habilita o botão Voltar.

O sistema desabilita para edição o campo Data Criação.

O sistema desabilita para edição o campo Data Alteração.

O sistema desabilita para edição o campo Código.

O sistema carrega a data corrente no campo Data Criação (R17).

O sistema carrega a data corrente no campo Data Alteração (E3).

O usuário preenche o campo nome.

O usuário pressiona o botão Salvar. (A1).

O caso de uso é finalizado.

Fluxos Alternativos

A1: Botão Voltar pressionado.

O usuário pressiona o botão voltar.

O sistema retorna a tela anterior.

Fluxos de Exceção

E3. Campo Data Alteração

Ao editar um registro o campo data alteração recebe a data corrente/atual.

UC005 – Manter Pais

Controle do Documento

Versão	Autor	Data	Descrição
1.0	Maicon Fang	12/08/2013	Elaboração
1.1	Maicon Fang	16/08/2013	Correções
2.0	Maicon Fang	13/12/2013	Correções

Descrição

Este caso de uso serve para fazer o cadastramento de um País

Data View

DV1 - Tela de cadastramento de País

Cadastro de País

Código: automático

Nome:

DDI:

Data Cadastro: Dec 11, 2013

Data Alteração: Dec 11, 2013

Pré-condições

Este caso de uso pode iniciar somente se:

O sistema tiver executado o UC011 - Login

O usuário possuir permissão para cadastrar um país.

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

Ter salvo os dados do País.

Ator Primário

Administrador

Fluxo de Eventos Principal

O sistema apresenta a tela (DV1)

O sistema habilita o botão Cadastrar.

O sistema habilita o botão Voltar.

O sistema desabilita para edição o campo Data Criação.

O sistema desabilita para edição o campo Data Alteração.

O sistema desabilita para edição o campo Código.

O sistema carrega a data corrente no campo Data Criação (R17).

O sistema carrega a data corrente no campo Data Alteração (E3).

O usuário preenche o campo nome.

- O usuário preenche o campo DDI.
- O usuário pressiona o botão Salvar. (A1).
- O caso de uso é finalizado.

Fluxos Alternativos

- A1: Botão Voltar pressionado.
- O usuário pressiona o botão voltar.
- O sistema retorna a tela anterior.

Fluxos de Exceção

E3. Campo Data Alteração

Ao editar um registro o campo data alteração recebe a data corrente/atual.

UC006 – Manter Estado

Controle do Documento

Versão	Autor	Data	Descrição
1.0	Maicon Fang	12/08/2013	Elaboração
1.1	Maicon Fang	16/08/2013	Correções
2.0	Maicon Fang	13/12/2013	Correções

Descrição

Este caso de uso serve para fazer o cadastramento de um Estado

Data View

DV1 - Tela de cadastramento de Estado



The screenshot shows a web form titled 'Cadastramento de Estado'. The form contains the following fields and controls:

- Código:** A text field with the value 'automático'.
- Nome:** A text field.
- UF:** A text field.
- País:** A dropdown menu with the selected option 'Selecione'.
- Data Cadastro:** A text field with the value '11/12/2013'.
- Data Alteração:** A text field with the value '11/12/2013'.
- Buttons:** Two buttons at the bottom right: 'cadastrar' and 'Voltar'.

Pré-condições

Este caso de uso pode iniciar somente se:

O sistema tiver executado o UC011 - Login

O usuário possuir permissão para cadastrar um estado.

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

Ter salvo os dados do Estado.

Ator Primário

Administrador

Fluxo de Eventos Principal

- O sistema apresenta a tela (DV1)
- O sistema habilita o botão Cadastrar.
- O sistema habilita o botão Voltar.
- O sistema carrega os países no combobox pais.
- O sistema desabilita para edição o campo Data Criação.
- O sistema desabilita para edição o campo Data Alteração.
- O sistema desabilita para edição o campo Código.
- O sistema carrega a data corrente no campo Data Criação (R17).
- O sistema carrega a data corrente no campo Data Alteração (E3).
- O usuário preenche o campo nome.
- O usuário preenche o campo UF.
- O usuário seleciona o pais no combobox o pais.
- O usuário pressiona o botão Salvar. (A1).
- O caso de uso é finalizado.

Fluxos Alternativos

- A1: Botão Voltar pressionado.
- O usuário pressiona o botão voltar.
- O sistema retorna a tela anterior.

Fluxos de Exceção

E3. Campo Data Alteração

Ao editar um registro o campo data alteração recebe a data corrente/atual.

UC007 – Manter Cidade

Controle do Documento

Versão	Autor	Data	Descrição
1.0	Maicon Fang	12/08/2013	Elaboração
1.1	Maicon Fang	16/08/2013	Correções
2.0	Maicon Fang	13/12/2013	Correções

Descrição

Este caso de uso serve para fazer o cadastramento de uma Cidade.

Data View

DV1 - Tela de cadastramento de Cidade



Cadastro de Cidade

Código: automático

Nome:

DDD:

Estado:

Data Cadastro:

Data Alteração:

Pré-condições

Este caso de uso pode iniciar somente se:

O sistema tiver executado o UC011 - Login

O usuário possuir permissão para cadastrar uma cidade.

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

Ter salvo os dados da Cidade.

Ator Primário

Administrador

Fluxo de Eventos Principal

O sistema apresenta a tela (DV1)

O sistema habilita o botão Cadastrar.

O sistema habilita o botão Voltar.

O sistema carrega os estados no combobox estado.

O sistema desabilita para edição o campo Data Criação.

O sistema desabilita para edição o campo Data Alteração.

O sistema desabilita para edição o campo Código.

O sistema carrega a data corrente no campo Data Criação (R17).

O sistema carrega a data corrente no campo Data Alteração (E3).

O usuário preenche o campo nome.

- O usuário preenche o campo DDD.
- O usuário seleciona o estado no combobox o estado.
- O usuário pressiona o botão Salvar. (A1).
- O caso de uso é finalizado.

Fluxos Alternativos

- A1: Botão Voltar pressionado.
- O usuário pressiona o botão voltar.
- O sistema retorna a tela anterior.

Fluxos de Exceção

E3. Campo Data Alteração

Ao editar um registro o campo data alteração recebe a data corrente/atual.

UC008 – Manter Usuário

Controle do Documento

Versão	Autor	Data	Descrição
1.0	Maicon Fang	12/08/2013	Elaboração
1.1	Maicon Fang	16/08/2013	Correções
2.0	Maicon Fang	13/12/2013	Correções

Descrição

Este caso de uso serve para fazer o cadastramento de um usuário.

Data View

DV1 - Tela de cadastramento de usuário

Cadastro de Usuário

Matricula:

Nome:

Login:

E-Mail:

Senha:

Confirme a Senha:

Perfil:

Pré-condições

Este caso de uso pode iniciar somente se:

O sistema tiver executado o UC011 - Login

O usuário possuir permissão para cadastrar um usuário.

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

Ter salvo os dados do usuário.

Ator Primário

Administrador

Fluxo de Eventos Principal

- O sistema apresenta a tela (DV1)
- O sistema habilita o botão Cadastrar.
- O sistema habilita o botão Voltar.
- O sistema carrega os perfis no combobox perfil.
- O usuário preenche o campo matricula.
- O usuário preenche o campo nome.
- O usuário preenche o login.
- O usuário preenche o campo e-mail.
- O usuário preenche o campo senha.
- O usuário preenche o campo confirmar senha.
- O usuário seleciona o perfil no combobox perfil.
- O usuário pressiona o botão Salvar. (A1).
- O caso de uso é finalizado.

Fluxos Alternativos

- A1: Botão Voltar pressionado.
- O usuário pressiona o botão voltar.
- O sistema retorna a tela anterior.

Fluxos de Exceção

N/A

UC009 – Manter Perfil

Controle do Documento

Versão	Autor	Data	Descrição
1.0	Maicon Fang	12/08/2013	Elaboração
1.1	Maicon Fang	16/08/2013	Correções
2.0	Maicon Fang	13/12/2013	Correções

Descrição

Este caso de uso serve para fazer o cadastramento de um Perfil.

Data View

DV1 - Tela de cadastramento de Perfil

Cadastro de Perfil

Código: automático

Nome:

Data Cadastro: 11/12/2013

Data Alteração: 11/12/2013

Pré-condições

Este caso de uso pode iniciar somente se:

O sistema tiver executado o UC011 - Login

O usuário possuir permissão para cadastrar um perfil.

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

Ter salvo os dados do Perfil.

Ator Primário

Administrador

Fluxo de Eventos Principal

O sistema apresenta a tela (DV1)

O sistema habilita o botão Cadastrar.

O sistema habilita o botão Voltar.

O sistema desabilita para edição o campo Data Criação.

O sistema desabilita para edição o campo Data Alteração.

O sistema desabilita para edição o campo Código.

O sistema carrega a data corrente no campo Data Criação (R17).

O sistema carrega a data corrente no campo Data Alteração (E3).

O usuário preenche o campo nome.

O usuário pressiona o botão Salvar. (A1).

O caso de uso é finalizado.

Fluxos Alternativos

A1: Botão Voltar pressionado.

O usuário pressiona o botão voltar.

O sistema retorna a tela anterior.

Fluxos de Exceção

E3. Campo Data Alteração

Ao editar um registro o campo data alteração recebe a data corrente/atual.

UC010 – Manter Atividade

Controle do Documento

Versão	Autor	Data	Descrição
1.0	Maicon Fang	12/08/2013	Elaboração
1.1	Maicon Fang	16/08/2013	Correções
2.0	Maicon Fang	13/12/2013	Correções

Descrição

Este caso de uso serve para fazer o cadastramento de uma atividade.

Data View

DV1 - Tela de cadastramento de atividade.



The screenshot shows a web form titled 'Cadastro de Atividade'. It contains the following fields and controls:

- Código:** A dropdown menu with 'automático' selected.
- Nome:** A text input field.
- Data Cadastro:** A date input field showing '11/12/2013'.
- Data Alteração:** A date input field showing '11/12/2013'.
- Buttons:** Two buttons at the bottom right, labeled 'cadastrat' and 'Voltar'.

Pré-condições

Este caso de uso pode iniciar somente se:

O sistema tiver executado o UC011 - Login

O usuário possuir permissão para cadastrar uma atividade.

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

Ter salvo os dados da atividade.

Ator Primário

Administrador

Fluxo de Eventos Principal

O sistema apresenta a tela (DV1)

O sistema habilita o botão Cadastrar.

O sistema habilita o botão Voltar.

- O sistema desabilita para edição o campo Data Criação.
- O sistema desabilita para edição o campo Data Alteração.
- O sistema desabilita para edição o campo Código.
- O sistema carrega a data corrente no campo Data Criação (R17).
- O sistema carrega a data corrente no campo Data Alteração (E3).
- O usuário preenche o campo nome.
- O usuário pressiona o botão Salvar. (A1).
- O caso de uso é finalizado.

Fluxos Alternativos

- A1: Botão Voltar pressionado.
- O usuário pressiona o botão voltar.
- O sistema retorna a tela anterior.

Fluxos de Exceção

E3. Campo Data Alteração

Ao editar um registro o campo data alteração recebe a data corrente/atual.

UC011 – Manter Login

Controle do Documento

Versão	Autor	Data	Descrição
--------	-------	------	-----------

1.0	Maicon Fang	12/08/2013	Elaboração
1.1	Maicon Fang	16/08/2013	Correções
2.0	Maicon Fang	13/12/2013	Correções

Descrição

Este caso de uso serve para acessar o sistema.

Data View

DV1 - Tela de login.

ACESSO RESTRITO

Login:

Senha:

Login

[Esqueci minha senha.](#)

Pré-condições

Este caso de uso pode iniciar somente se:

O usuário possuir permissão para acessar o sistema.

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

Ter permitido o acesso ao sistema.

Ator Primário

Administrador

Fluxo de Eventos Principal

O sistema apresenta a tela (DV1).

O sistema habilita o botão login.

O usuário preenche o campo login.

O usuário preenche o campo senha.

O usuário pressiona o botão Login. (E1).

O caso de uso é finalizado.

Fluxos Alternativos

A1: Botão Voltar pressionado.

O usuário pressiona o botão voltar.

O sistema retorna a tela anterior.

Fluxos de Exceção

E1. Login e senha incorretos.

O sistema exibe a mensagem “Não foi possível se autenticar. Login ou senha inválidos!”

UC500 – Exibir empresa/s na aplicação Android.

Controle do Documento

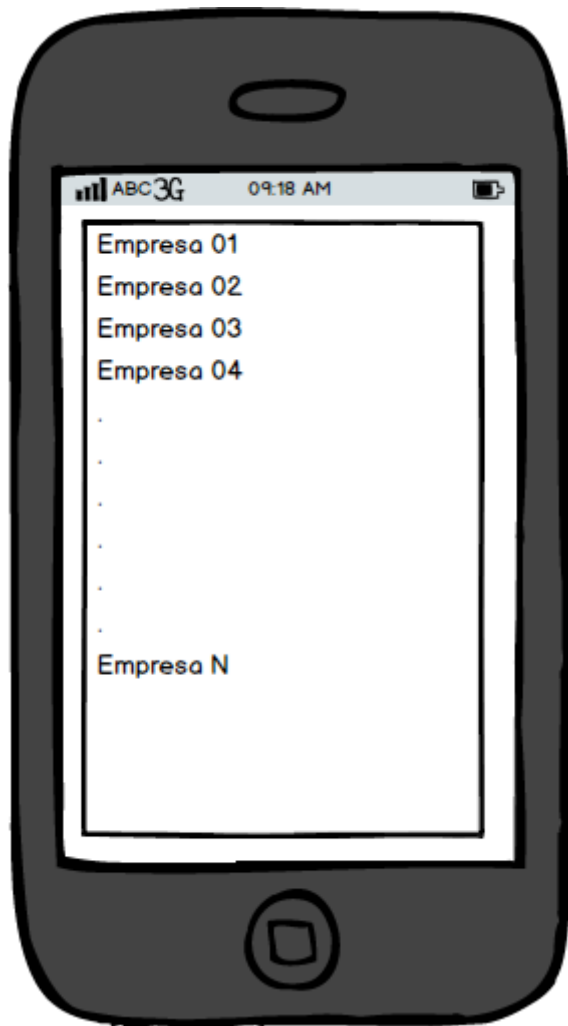
Versão	Autor	Data	Descrição
1.0	Maicon Fang	12/08/2013	Elaboração
1.1	Maicon Fang	16/08/2013	Correções
2.0	Maicon Fang	12/12/2013	Correções
2.1	Maicon Alexander Illes Fang	19/02/2014	Remoção do caso de uso 502

Descrição

Este caso de uso serve para exibir a/s empresa/s na aplicação Android.

Data Views

DV1 - Tela de exibição da empresa.



Pré-condições

Este caso de uso pode iniciar somente se:

Existir conexão com a internet.

A aplicação estiver previamente instalada no dispositivo móvel.

Ter acessado a aplicação.

Web Service estiver ativo.

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

Ter exibido a/s empresa/s.

Ator Primário

Usuário

Fluxo de Eventos Principal

Ao inicializar o aplicativo as empresas são carregadas (DV1), (R10).

O usuário seleciona a empresa (A1), (A2).

O sistema chama o Use Case 501

O caso de uso é finalizado.

Fluxos Alternativos

A1: Botão “Voltar” pressionado.

O sistema retorna para tela anterior.

O caso de uso é finalizado

A2: Botão “Home” pressionado.

O sistema retorna para tela principal do dispositivo móvel.

O caso de uso é finalizado

Fluxos de Exceção

Não aplicável.

UC501 – Exibir produto/s na aplicação Android.

Controle do Documento

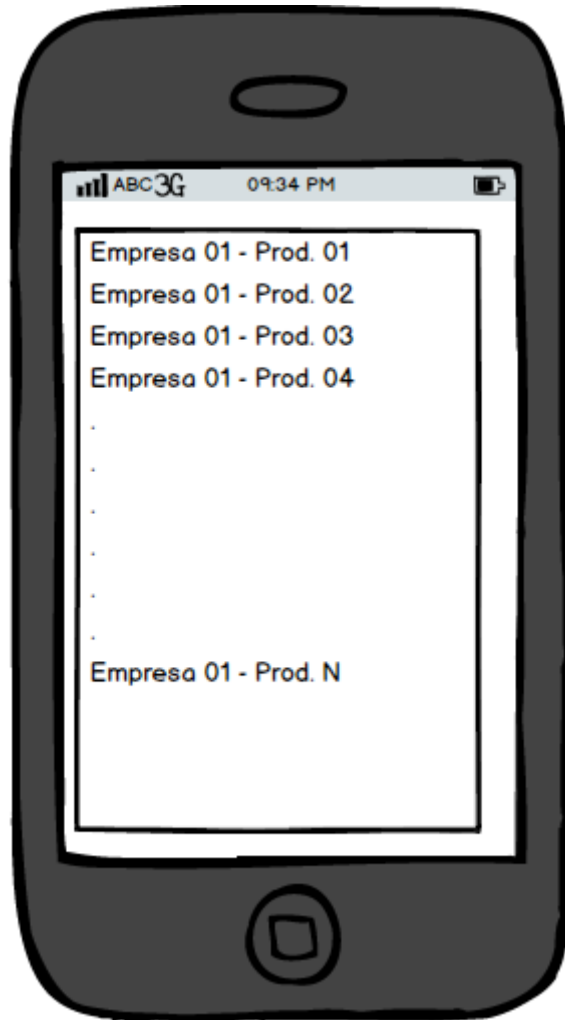
Versão	Autor	Data	Descrição
1.0	Maicon Fang	12/08/2013	Elaboração
1.1	Maicon Fang	16/08/2013	Correções
2.0	Maicon Fang	12/12/2013	Correções
2.1	Maicon Alexander Illes Fang	19/02/2014	Correções

Descrição

Este caso de uso serve para exibir os produtos na aplicação Android.

Data Views

DV2 - Tela de exibição dos produtos.



Pré-condições

Este caso de uso pode iniciar somente se:

A aplicação estiver previamente instalada no dispositivo móvel.

Ter acessado a aplicação.

Ter executado o Use Case 500.

Web Service estiver ativo

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

Ter exibido os produtos.

Ator Primário

Usuário

Fluxo de Eventos Principal

Ao clicar na empresa o/s produto/s são exibidos (DV1), (R4), (R11), (E1), (E2)

O usuário seleciona o produto.

O sistema chama o Use case 502

O caso de uso é finalizado.

Fluxos Alternativos

A1: Botão “Voltar” pressionado.

O sistema retorna para tela anterior.

O sistema apresenta a tela com as empresas cadastradas.

O caso de uso é finalizado.

A2: Botão “Home” pressionado.

O sistema retorna para tela principal do dispositivo móvel.

O caso de uso é finalizado.

Fluxos de Exceção

Não aplicável.

UC502 – Exibir detalhe de um produto na aplicação Android.

Controle do Documento

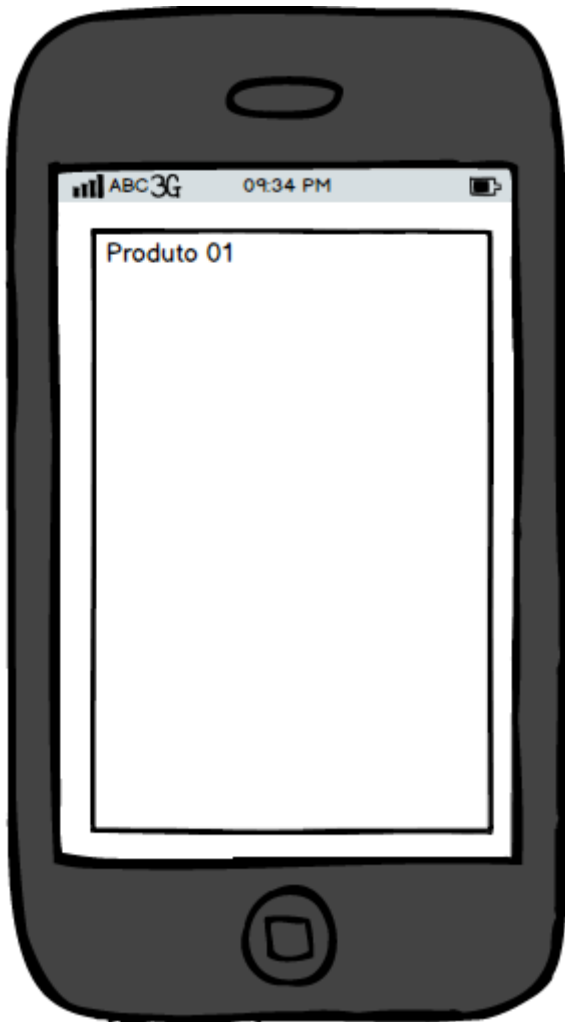
Versão	Autor	Data	Descrição
1.0	Maicon Fang	12/08/2013	Elaboração
1.1	Maicon Fang	16/08/2013	Correções
2.0	Maicon Fang	12/12/2013	Correções
2.1	Maicon Alexander Illes Fang	19/02/2014	Correções

Descrição

Este caso de uso serve para exibir um único produto na aplicação Android.

Data Views

DV1 - Tela de exibição do produto.



Pré-condições

Este caso de uso pode iniciar somente se:

Existir conexão com a internet.

A aplicação estiver previamente instalada no dispositivo móvel.

Ter acessado a aplicação.

Ter executado o Use Case 501.

O Web Service estiver ativo.

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

Ter exibido o produto.

Ator Primário

Usuário

Fluxo de Eventos Principal

Á aplicação exibe o produto selecionado (DV1), (E1), (E2)

O caso de uso é finalizado

Fluxos Alternativos

A1: Botão “Voltar” pressionado.

O sistema retorna para tela anterior.

O caso de uso é finalizado

A2: Botão “Home” pressionado.

O sistema retorna para tela principal do dispositivo móvel.

O caso de uso é finalizado

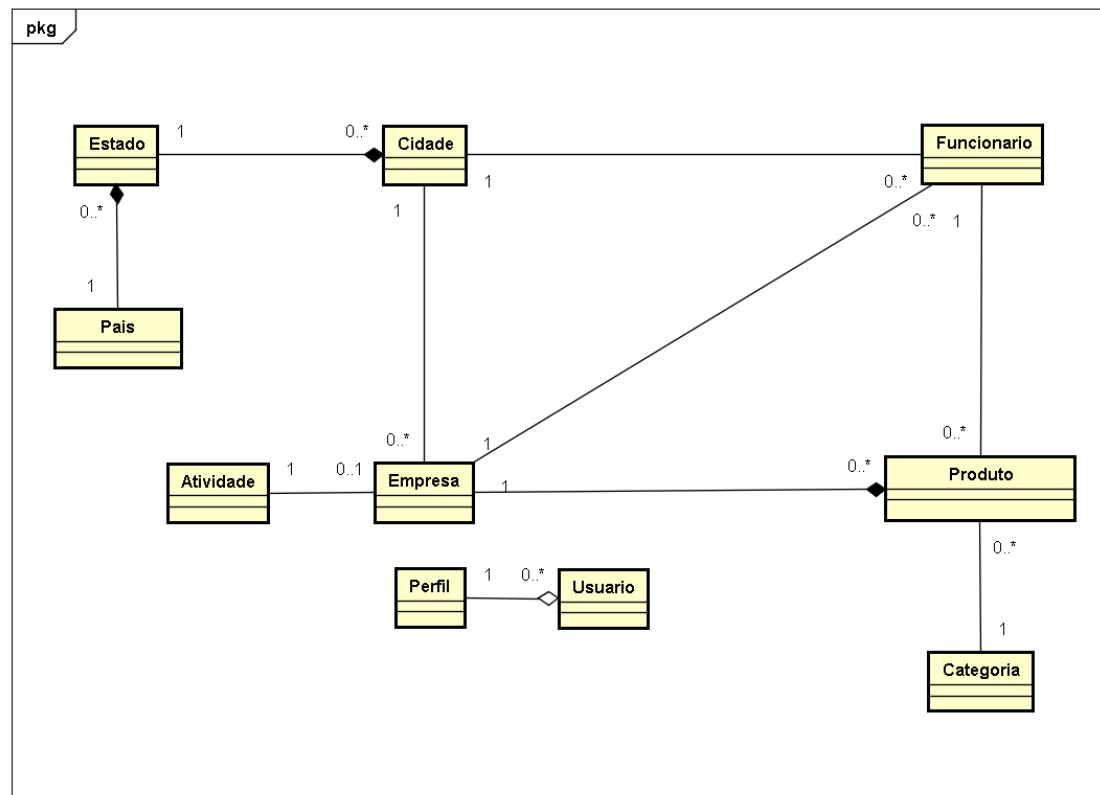
Fluxos de Exceção

Não aplicável.

APÊNDICE G - 7 - Template Modelo de Objetos Negociais

Versão 2.1

Diagrama de Classes sem atributos e com Relacionamentos



APÊNDICE H - 8 - Template Casos de Uso Negociais

Diagrama de Caso de Uso e especificação detalhada dos casos de uso.

Versão 2.1

Histórico da Revisão

Data	Versão	Descrição	Autor
10/08/2013	1.0	Versão inicial	Maicon Alexander Illes Fang
11/12/2013	2.0	Versão secundaria	Maicon Alexander Illes Fang
19/02/2014	2.1	Ajustes e inserção de novos casos de usos.	Maicon Alexander Illes Fang

UC001 – Manter Empresa

Controle do Documento

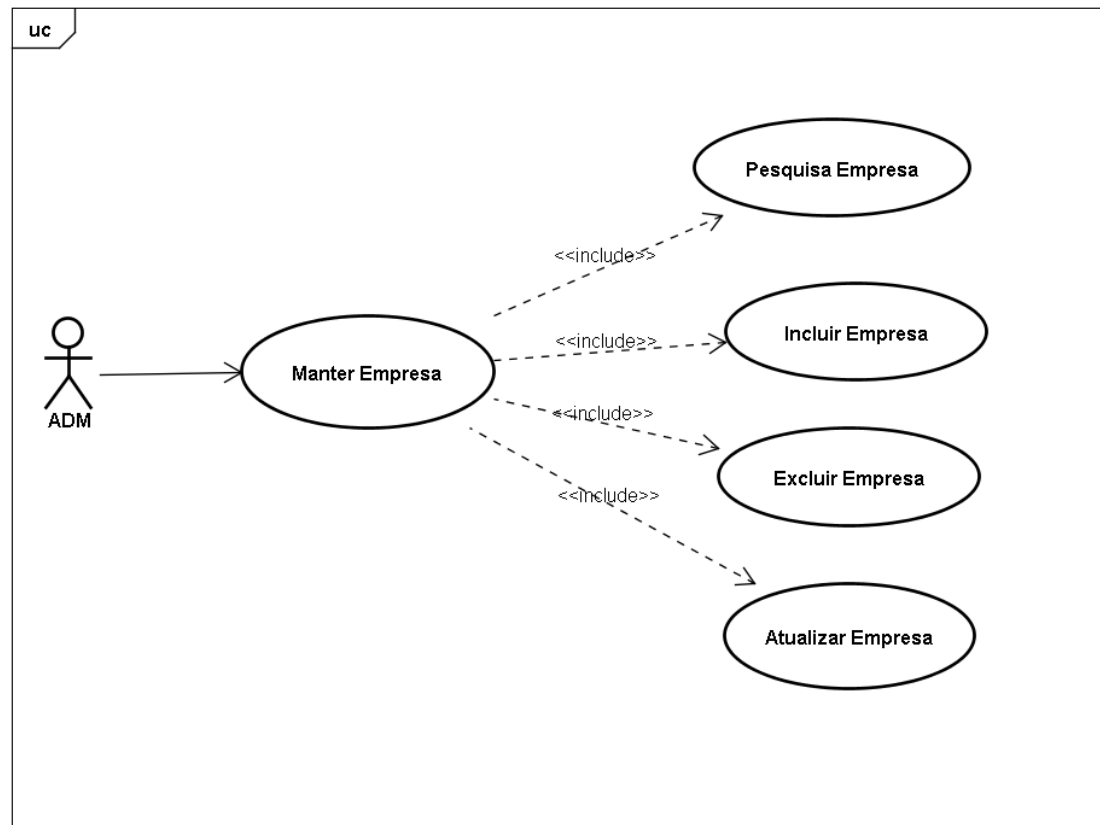
Versão	Autor	Data	Descrição
1.0	Maicon Fang	12/08/2013	Elaboração
1.1	Maicon Fang	16/08/2013	Correções
2.0	Maicon Fang	13/12/2013	Correções

Descrição

Este caso de uso serve para cadastrar uma empresa.

Diagrama de Caso de Uso

DCU - Empresa.



Pré-condições

Este caso de uso pode iniciar somente se:

O sistema tiver executado o UC011 - Login

O usuário possuir permissão para cadastrar uma empresa.

O usuário tiver cadastrado uma cidade e atividade.

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

Ter salvo os dados da empresa.

Ator Primário

Administrador

Fluxo de Eventos Principal

O sistema apresenta a tela (DV1)

O sistema habilita o botão Cadastrar.

O sistema carrega a data corrente no campo Data Criação (R17).

O sistema carrega a data corrente no campo Data Alteração (E1).

O sistema desabilita para edição o campo Data Criação.

O sistema desabilita para edição o campo Data Alteração.

O sistema desabilita para edição o campo Código.

O sistema carrega a lista de Cidades cadastradas.

O sistema carrega a lista de Atividades cadastradas.

O usuário preenche o campo Nome.

O usuário seleciona o campo Cidade.

O usuário seleciona o campo Atividade.

O usuário seleciona o campo Foto.

O usuário preenche o campo CNPJ (R11).

O usuário preenche o campo IE.
O usuário preenche o campo CEP.
O usuário preenche o campo Logradouro.
O usuário preenche o campo Número.
O usuário preenche o campo Complemento.
O usuário preenche o campo Bairro.
O usuário preenche o campo Telefone.
O usuário preenche o campo Contato.
O usuário preenche o campo Site.
O usuário pressiona o botão Cadastrar. (A3)
O caso de uso é finalizado.

Fluxos Alternativos

A3: Botão Voltar pressionado.
O usuário pressiona o botão voltar.
O sistema retorna a tela anterior.

Fluxos de Exceção

E1. Campo Data Alteração.
Ao editar um registro o campo data alteração recebe a data corrente/atual.

UC002 – Manter Funcionário

Controle do Documento

Versão	Autor	Data	Descrição
1.0	Maicon Fang	12/08/2013	Elaboração
1.1	Maicon Fang	16/08/2013	Correções
2.0	Maicon Fang	13/12/2013	Correções

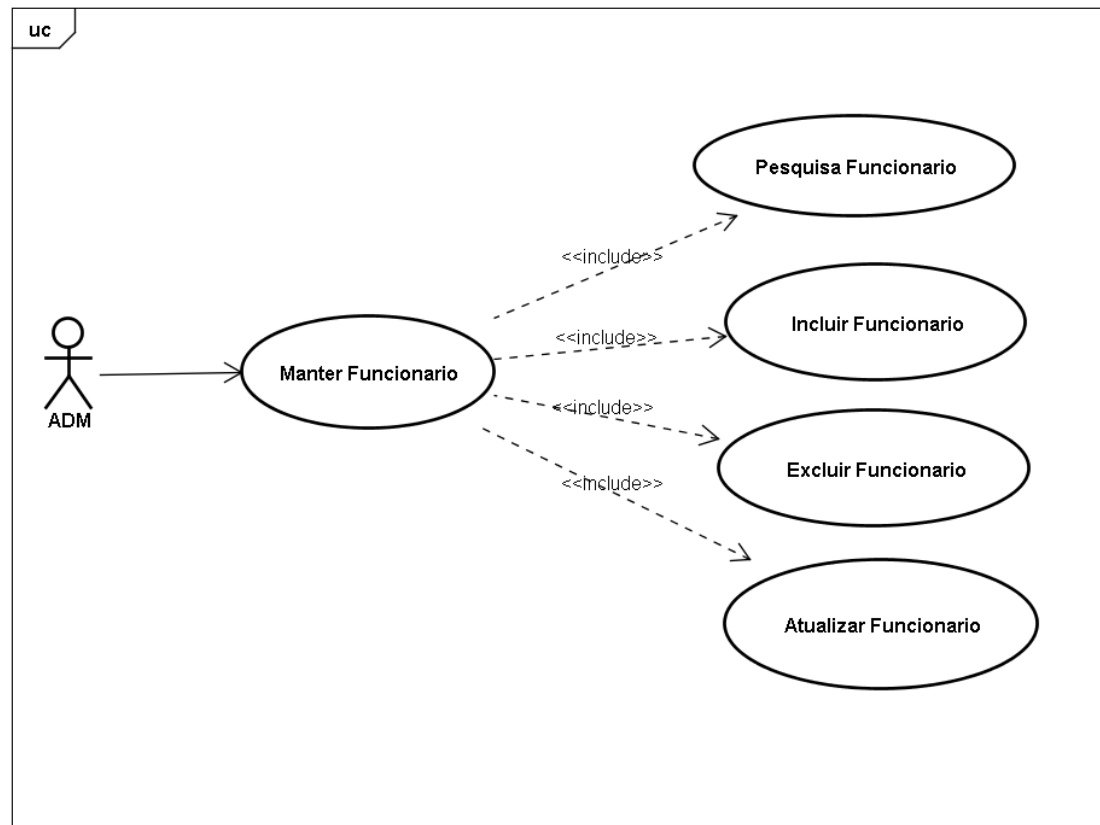
Especificação de Caso de Uso

Descrição

Este caso de uso serve para cadastrar um funcionário.

Diagrama de Caso de Uso

DCU – Funcionário.



Pré-condições

Este caso de uso pode iniciar somente se:

O sistema tiver executado o UC011 - Login

O usuário possuir permissão para cadastrar um funcionário.

O usuário tiver cadastrado uma cidade e empresa.

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

Ter salvo os dados do Funcionário.

Ator Primário

Administrador

Fluxo de Eventos Principal

O sistema apresenta a tela (DV1)

O sistema habilita o botão Cadastrar.

O sistema habilita o botão Voltar.

O sistema desabilita para edição o campo Data Criação (R17).

O sistema desabilita para edição o campo Data Alteração (E).

O sistema desabilita para edição o campo Código.

O sistema carrega a data corrente no campo Data Criação (A2).

O sistema carrega a data corrente no campo Data Alteração.

O sistema carrega uma lista de Empresas no combo Box Empresa.

O sistema carrega uma lista de Cidades no combo Box Cidade.

O usuário preenche o campo Nome.

O usuário seleciona o campo Cidade.

O usuário seleciona o campo Empresa.

O usuário seleciona o campo Foto.
O usuário preenche o campo CPF (R21).
O usuário preenche o campo RG.
O usuário preenche o campo Data Nascimento.
O usuário preenche o campo CEP.
O usuário preenche o campo Logradouro.
O usuário preenche o campo Bairro.
O usuário preenche o campo Número.
O usuário preenche o campo Complemento.
O usuário preenche o campo Contato.
O usuário preenche o campo E-mail.
O usuário preenche o campo Rede Social.
O usuário preenche o campo Ativo.
O usuário pressiona o botão Cadastrar. (A1)
O caso de uso é finalizado.

Fluxos Alternativos

A1: Botão Voltar pressionado.
O usuário pressiona o botão voltar.
O sistema retorna a tela anterior.

Fluxos de Exceção

E1. Campo Data Alteração.

Ao editar um registro o campo data alteração recebe a data corrente/atual.

UC003 – Manter Produto

Controle do Documento

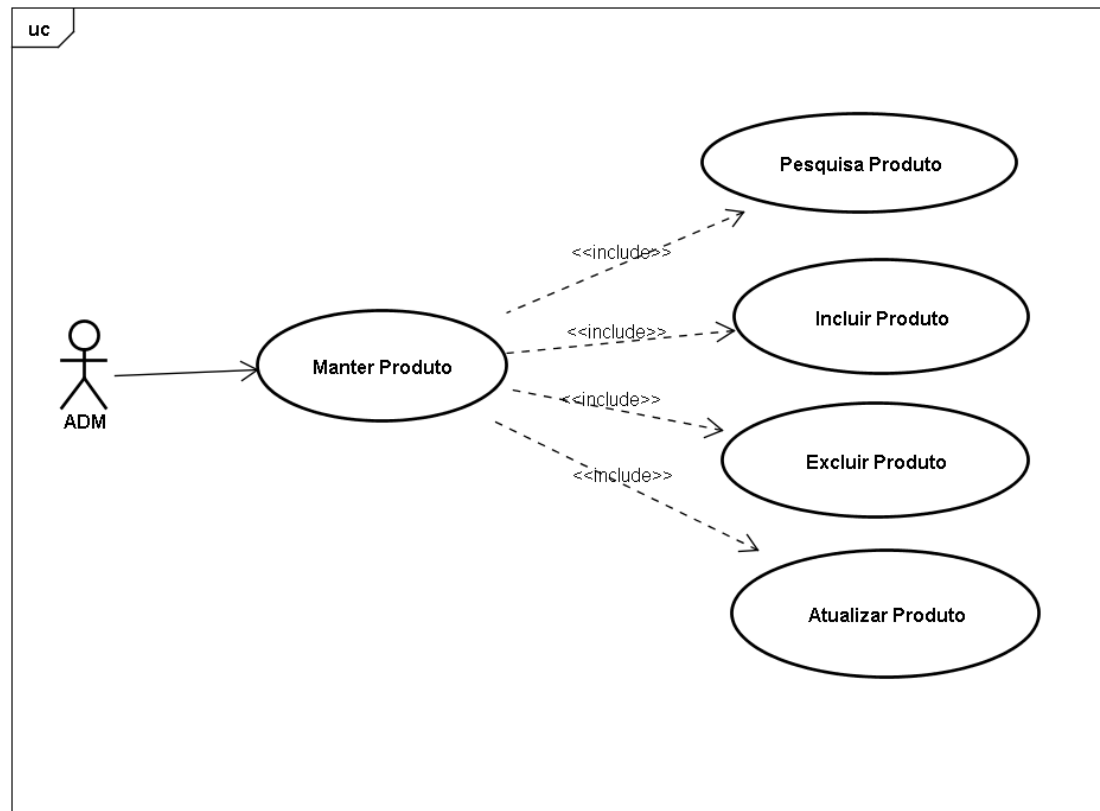
Versão	Autor	Data	Descrição
1.0	Maicon Fang	12/08/2013	Elaboração
1.1	Maicon Fang	16/08/2013	Correções
2.0	Maicon Fang	13/12/2013	Correções

Descrição

Este caso de uso serve para cadastrar um produto

Diagrama de Caso de Uso

DCU – Produto.



Pré-condições

Este caso de uso pode iniciar somente se:

O sistema tiver executado o UC011 - Login

O usuário possuir permissão para cadastrar um Produto.

O sistema deve ter um Funcionário, Empresa e Categoria cadastrada.

Pós-condições

Após o fim normal deste caso de uso o sistema deve:
Ter salvo os dados do Produto.

Ator Primário

Funcionário.

Fluxo de Eventos Principal

- O sistema apresenta a tela (DV1)
- O sistema habilita o botão Cadastrar.
- O sistema habilita o botão Voltar.
- O sistema desabilita para edição o campo Data Criação.
- O sistema desabilita para edição o campo Data Alteração.
- O sistema desabilita para edição o campo Código.
- O sistema carrega a data corrente no campo Data Criação (R17).
- O sistema carrega a data corrente no campo Data Alteração (E3).
- O sistema carrega o campo ativo preenchido e desabilitado (E2).
- O sistema carrega o combo Box Categoria.
- O sistema carrega o combo Box Empresa.
- O sistema carrega o combo Box Funcionário.
- O usuário seleciona o campo Foto do Produto.

O usuário seleciona o campo Categoria do Produto.
O usuário preenche o campo Nome.
O usuário preenche o campo Descrição.
O usuário preenche o campo Especificações Técnicas.
O usuário preenche o campo Quantidade.
O usuário preenche o campo Preço Normal.
O usuário preenche o campo Preço Promoção (R19).
O usuário preenche o campo Data Validade (R20).
O usuário pressiona o botão Salvar. (A1).
O caso de uso é finalizado.

Fluxos Alternativos

A1: Botão Voltar pressionado.
O usuário pressiona o botão voltar.
O sistema retorna a tela anterior.

Fluxos de Exceção

E1. Lista sem produto cadastrada
O sistema exibe a lista sem produtos cadastrados
O sistema desabilita os botões Editar, Excluir

E2. Campo ativo editar registro

O sistema habilita o campo com as opções ativo e inativo

E3. Campo Data Alteração

Ao editar um registro o campo data alteração recebe a data corrente/atual.

UC004 – Manter Categoria

Controle do Documento

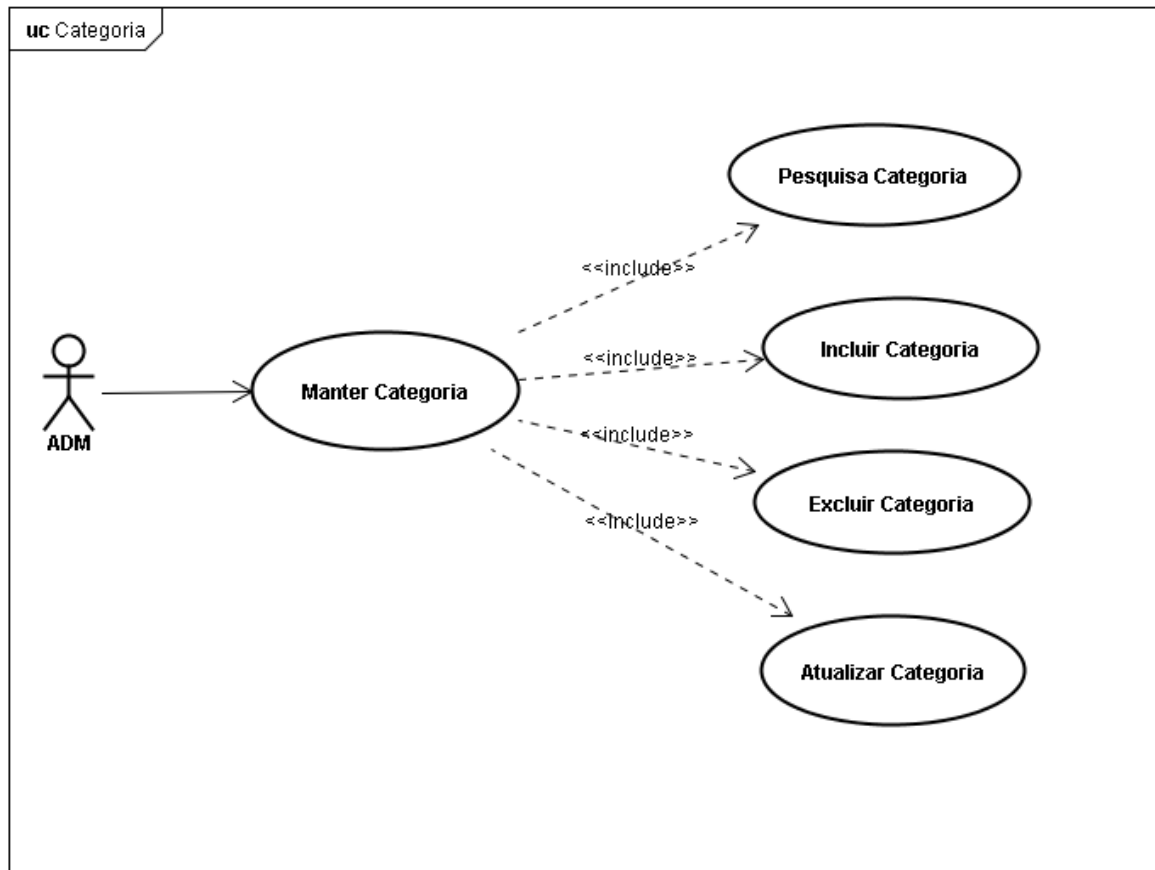
Versão	Autor	Data	Descrição
1.0	Maicon Fang	12/08/2013	Elaboração
1.1	Maicon Fang	16/08/2013	Correções
2.0	Maicon Fang	13/12/2013	Correções

Descrição

Este caso de uso serve para fazer o cadastramento de uma categoria.

Diagrama de Caso de Uso

DCU – Categoria.



Pré-condições

Este caso de uso pode iniciar somente se:

O sistema tiver executado o UC011 - Login

O usuário possuir permissão para cadastrar uma categoria.

Pós-condições

Após o fim normal deste caso de uso o sistema deve:
Ter salvo os dados da categoria.

Ator Primário

Administrador

Fluxo de Eventos Principal

- O sistema apresenta a tela (DV1)
- O sistema habilita o botão Cadastrar.
- O sistema habilita o botão Voltar.
- O sistema desabilita para edição o campo Data Criação.
- O sistema desabilita para edição o campo Data Alteração.
- O sistema desabilita para edição o campo Código.
- O sistema carrega a data corrente no campo Data Criação (R17).
- O sistema carrega a data corrente no campo Data Alteração (E3).
- O usuário preenche o campo nome.
- O usuário pressiona o botão Salvar. (A1).
- O caso de uso é finalizado.

Fluxos Alternativos

A1: Botão Voltar pressionado.

O usuário pressiona o botão voltar.

O sistema retorna a tela anterior.

Fluxos de Exceção

E3. Campo Data Alteração

Ao editar um registro o campo data alteração recebe a data corrente/atual.

UC005 – Manter Pais

Controle do Documento

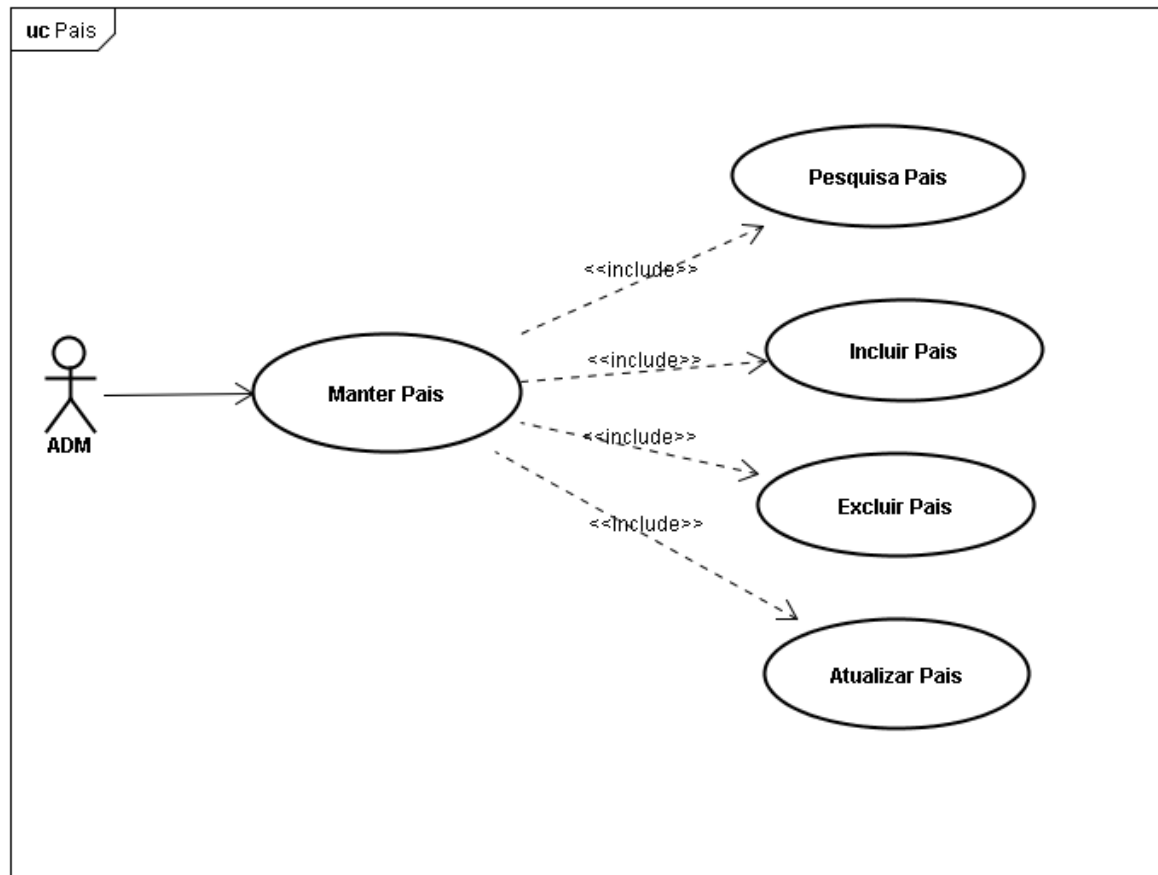
Versão	Autor	Data	Descrição
1.0	Maicon Fang	12/08/2013	Elaboração
1.1	Maicon Fang	16/08/2013	Correções
2.0	Maicon Fang	13/12/2013	Correções

Descrição

Este caso de uso serve para fazer o cadastramento de um pais.

Diagrama de Caso de Uso

DCU – Pais.



Pré-condições

Este caso de uso pode iniciar somente se:

O sistema tiver executado o UC011 - Login

O usuário possuir permissão para cadastrar um país.

Pós-condições

Após o fim normal deste caso de uso o sistema deve:
Ter salvo os dados do país.

Ator Primário

Administrador

Fluxo de Eventos Principal

- O sistema apresenta a tela (DV1)
- O sistema habilita o botão Cadastrar.
- O sistema habilita o botão Voltar.
- O sistema desabilita para edição o campo Data Criação.
- O sistema desabilita para edição o campo Data Alteração.
- O sistema desabilita para edição o campo Código.
- O sistema carrega a data corrente no campo Data Criação (R17).
- O sistema carrega a data corrente no campo Data Alteração (E3).
- O usuário preenche o campo nome.
- O usuário preenche o campo DDI.
- O usuário pressiona o botão Salvar. (A1).
- O caso de uso é finalizado.

Fluxos Alternativos

A1: Botão Voltar pressionado.

O usuário pressiona o botão voltar.

O sistema retorna a tela anterior.

Fluxos de Exceção

E3. Campo Data Alteração

Ao editar um registro o campo data alteração recebe a data corrente/atual.

UC006 – Manter Estado

Controle do Documento

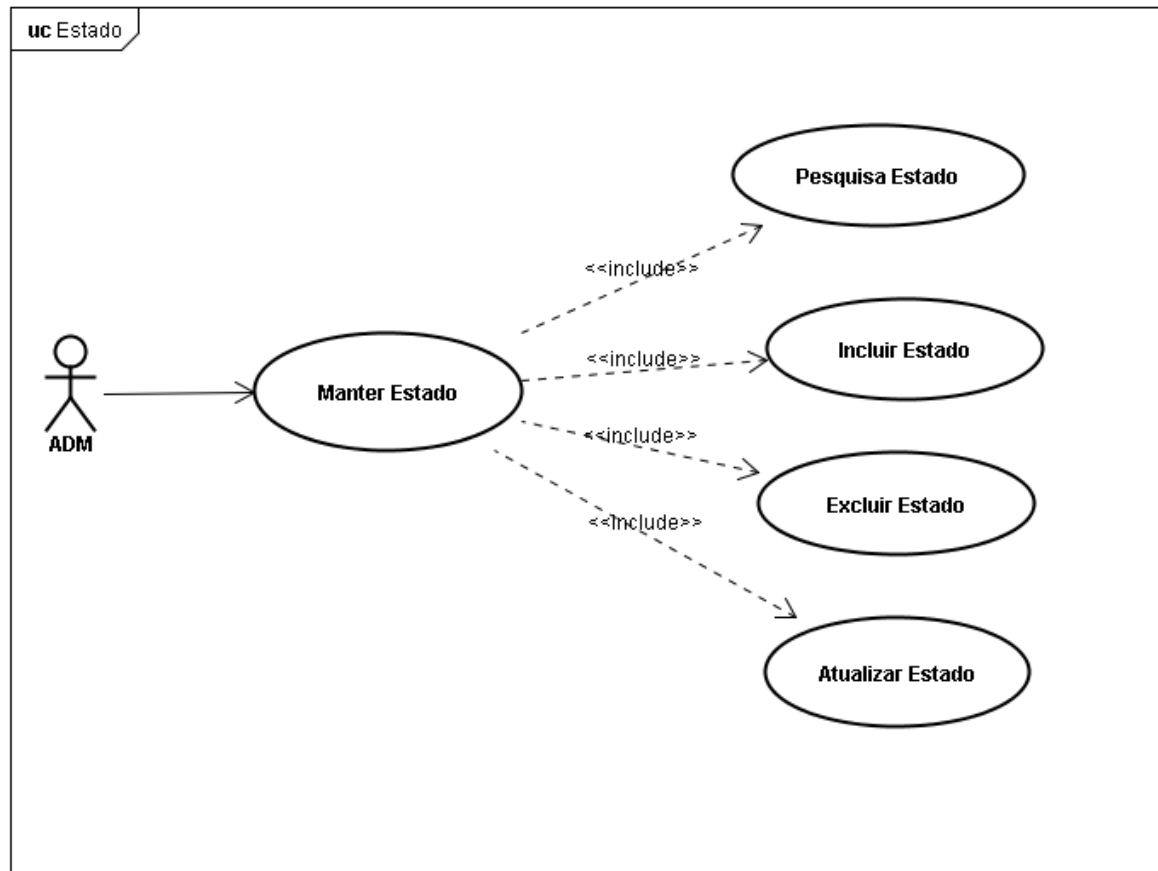
Versão	Autor	Data	Descrição
1.0	Maicon Fang	12/08/2013	Elaboração
1.1	Maicon Fang	16/08/2013	Correções
2.0	Maicon Fang	13/12/2013	Correções

Descrição

Este caso de uso serve para fazer o cadastramento de um estado.

Diagrama de Caso de Uso

DCU – Estado.



Pré-condições

Este caso de uso pode iniciar somente se:

O sistema tiver executado o UC011 - Login

O usuário possuir permissão para cadastrar um estado.

Pós-condições

Após o fim normal deste caso de uso o sistema deve:
Ter salvo os dados do estado.

Ator Primário

Administrador

Fluxo de Eventos Principal

- O sistema apresenta a tela (DV1)
- O sistema habilita o botão Cadastrar.
- O sistema habilita o botão Voltar.
- O sistema carrega os países no combobox pais.
- O sistema desabilita para edição o campo Data Criação.
- O sistema desabilita para edição o campo Data Alteração.
- O sistema desabilita para edição o campo Código.
- O sistema carrega a data corrente no campo Data Criação (R17).
- O sistema carrega a data corrente no campo Data Alteração (E3).
- O usuário preenche o campo nome.
- O usuário preenche o campo UF.
- O usuário seleciona o pais no combobox o pais.
- O usuário pressiona o botão Salvar. (A1).

O caso de uso é finalizado.

Fluxos Alternativos

A1: Botão Voltar pressionado.

O usuário pressiona o botão voltar.

O sistema retorna a tela anterior.

Fluxos de Exceção

E3. Campo Data Alteração

Ao editar um registro o campo data alteração recebe a data corrente/atual.

UC007 – Manter Cidade

Controle do Documento

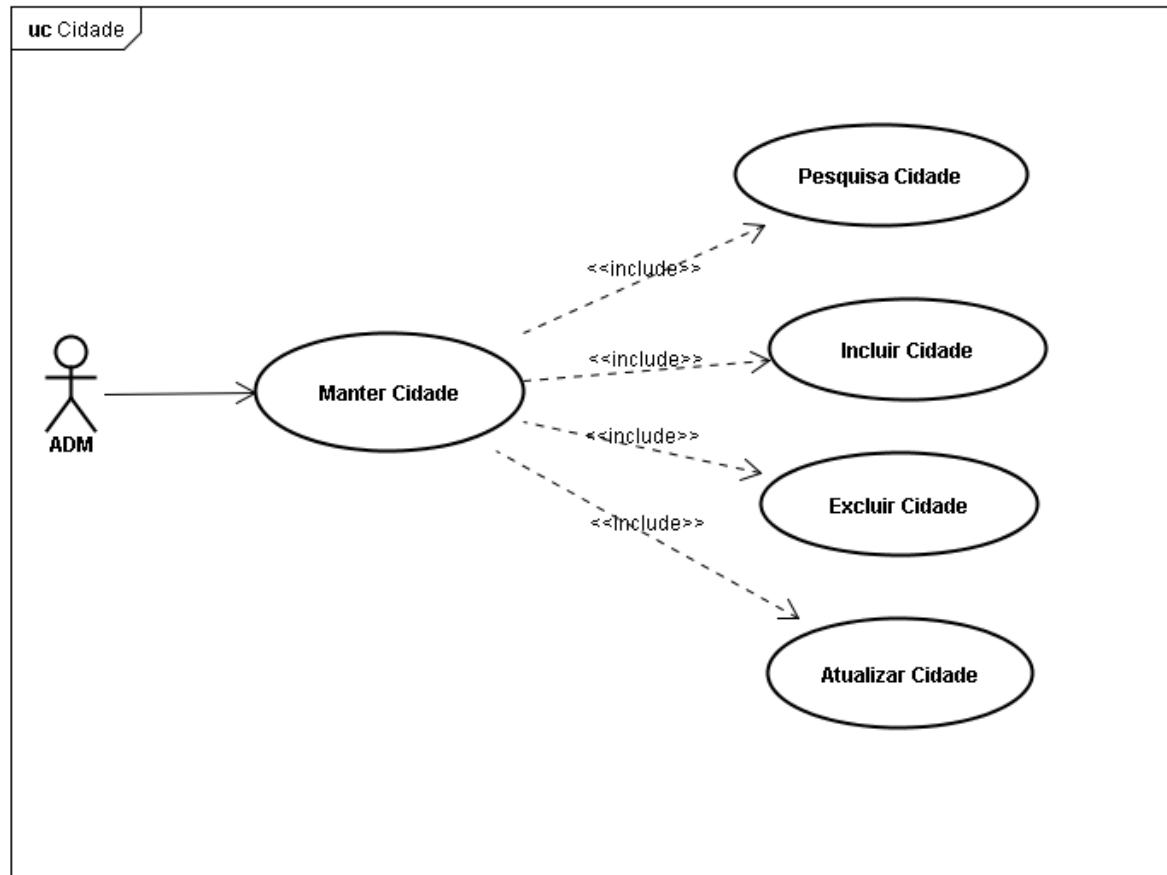
Versão	Autor	Data	Descrição
1.0	Maicon Fang	12/08/2013	Elaboração
1.1	Maicon Fang	16/08/2013	Correções
2.0	Maicon Fang	13/12/2013	Correções

Descrição

Este caso de uso serve para fazer o cadastramento de uma cidade.

Diagrama de Caso de Uso

DCU – Cidade.



Pré-condições

Este caso de uso pode iniciar somente se:

O sistema tiver executado o UC011 - Login

O usuário possuir permissão para cadastrar uma cidade.

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

Ter salvo os dados da cidade.

Ator Primário

Administrador

Fluxo de Eventos Principal

O sistema apresenta a tela (DV1)

O sistema habilita o botão Cadastrar.

O sistema habilita o botão Voltar.

O sistema carrega os estados no combobox estado.

O sistema desabilita para edição o campo Data Criação.

O sistema desabilita para edição o campo Data Alteração.

O sistema desabilita para edição o campo Código.

O sistema carrega a data corrente no campo Data Criação (R17).

O sistema carrega a data corrente no campo Data Alteração (E3).

O usuário preenche o campo nome.

O usuário preenche o campo DDD.

O usuário seleciona o estado no combobox o estado.

O usuário pressiona o botão Salvar. (A1).

O caso de uso é finalizado.

Fluxos Alternativos

A1: Botão Voltar pressionado.

O usuário pressiona o botão voltar.

O sistema retorna a tela anterior.

Fluxos de Exceção

E3. Campo Data Alteração

Ao editar um registro o campo data alteração recebe a data corrente/atual.

UC008 – Manter Usuário

Controle do Documento

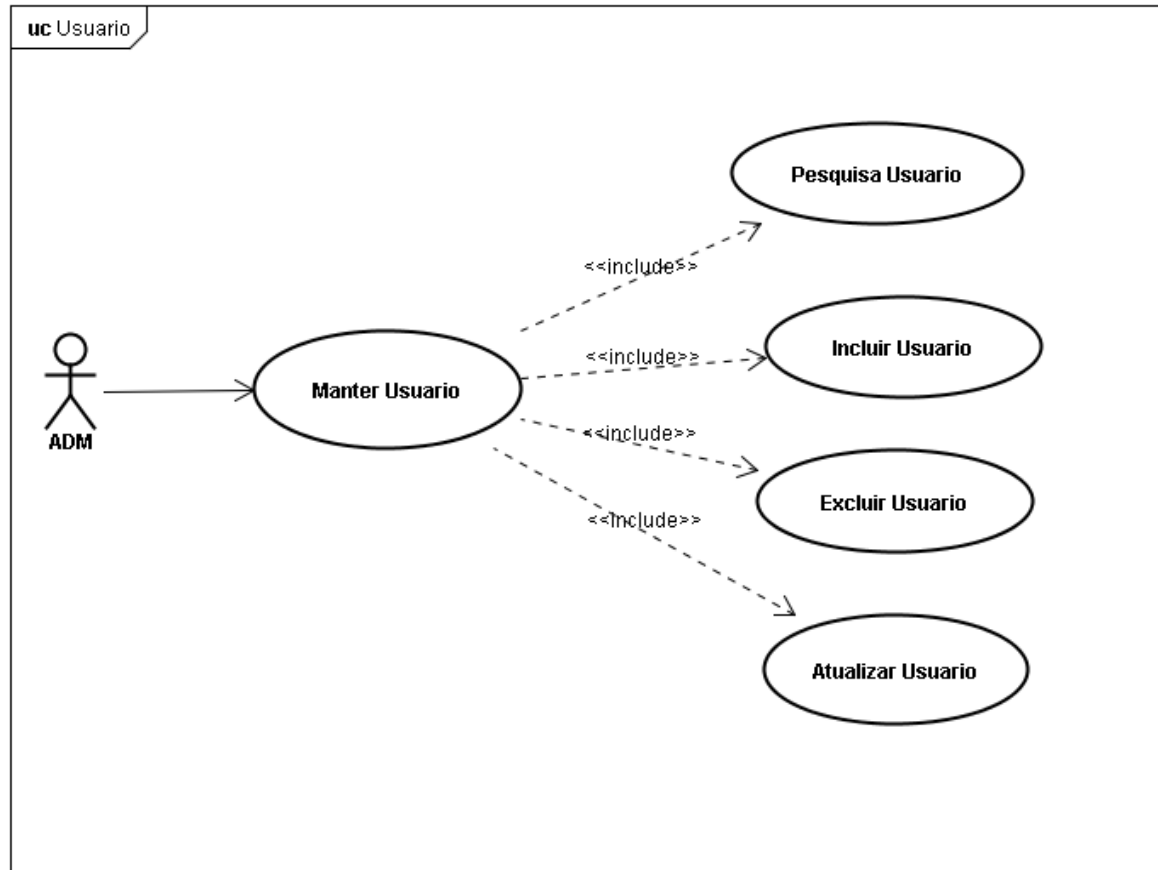
Versão	Autor	Data	Descrição
1.0	Maicon Fang	12/08/2013	Elaboração
1.1	Maicon Fang	16/08/2013	Correções
2.0	Maicon Fang	13/12/2013	Correções

Descrição

Este caso de uso serve para fazer o cadastramento de um usuário.

Diagrama de Caso de Uso

DCU – Usuário.



Pré-condições

Este caso de uso pode iniciar somente se:

O sistema tiver executado o UC011 - Login

O usuário possuir permissão para cadastrar um usuário.

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

Ter salvo os dados da usuário.

Ator Primário

Administrador

Fluxo de Eventos Principal

O sistema apresenta a tela (DV1)

O sistema habilita o botão Cadastrar.

O sistema habilita o botão Voltar.

O sistema carrega os perfis no combobox perfil.

O usuário preenche o campo matricula.

O usuário preenche o campo nome.

O usuário preenche o login.

O usuário preenche o campo e-mail.

O usuário preenche o campo senha.

O usuário preenche o campo confirmar senha.

O usuário seleciona o perfil no combobox perfil.

O usuário pressiona o botão Salvar. (A1).

O caso de uso é finalizado.

Fluxos Alternativos

A1: Botão Voltar pressionado.

O usuário pressiona o botão voltar.

O sistema retorna a tela anterior.

Fluxos de Exceção

N/A

UC009 – Manter Perfil

Controle do Documento

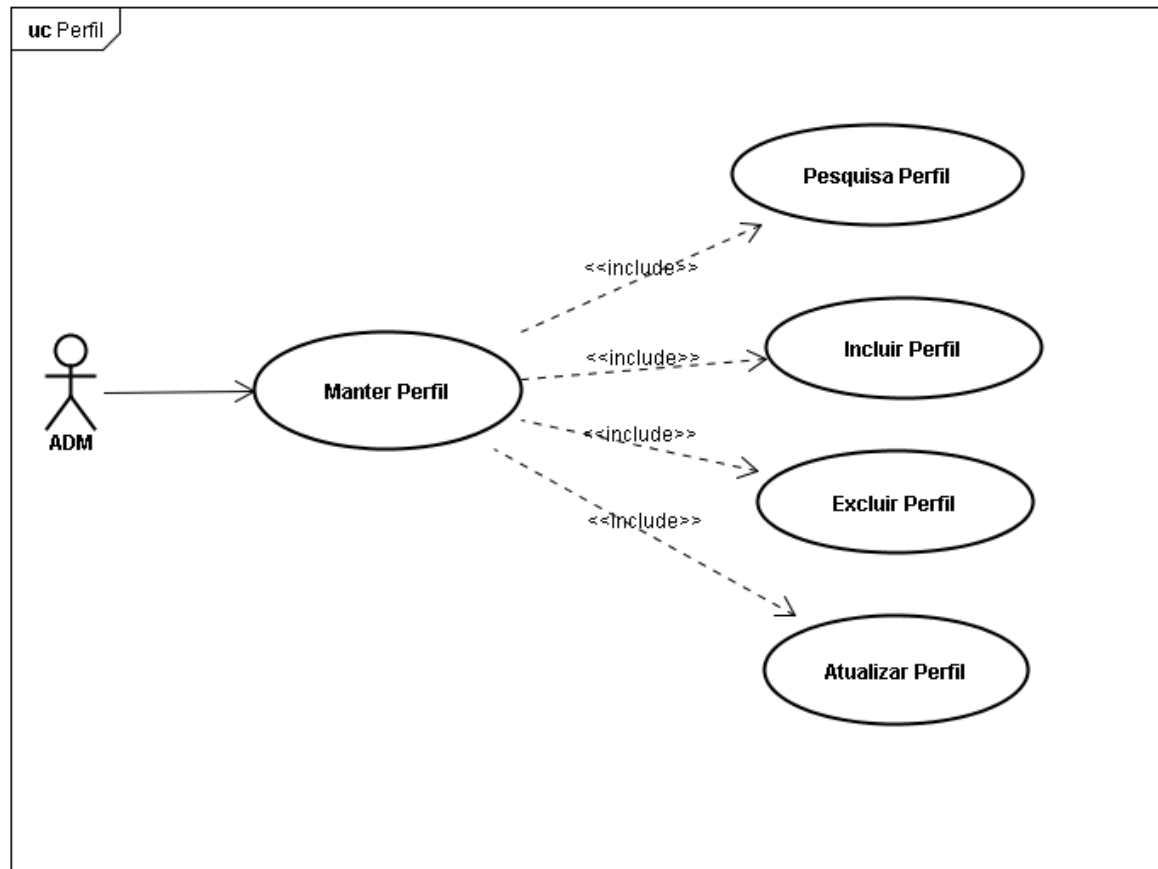
Versão	Autor	Data	Descrição
1.0	Maicon Fang	12/08/2013	Elaboração
1.1	Maicon Fang	16/08/2013	Correções
2.0	Maicon Fang	13/12/2013	Correções

Descrição

Este caso de uso serve para fazer o cadastramento de um perfil.

Diagrama de Caso de Uso

DCU – Perfil.



Pré-condições

Este caso de uso pode iniciar somente se:

O sistema tiver executado o UC011 - Login

O usuário possuir permissão para cadastrar um perfil.

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

Ter salvo os dados do perfil.

Ator Primário

Administrador

Fluxo de Eventos Principal

O sistema apresenta a tela (DV1)

O sistema habilita o botão Cadastrar.

O sistema habilita o botão Voltar.

O sistema desabilita para edição o campo Data Criação.

O sistema desabilita para edição o campo Data Alteração.

O sistema desabilita para edição o campo Código.

O sistema carrega a data corrente no campo Data Criação (R17).

O sistema carrega a data corrente no campo Data Alteração (E3).

O usuário preenche o campo nome.

O usuário pressiona o botão Salvar. (A1).

O caso de uso é finalizado.

Fluxos Alternativos

A1: Botão Voltar pressionado.

O usuário pressiona o botão voltar.

O sistema retorna a tela anterior.

Fluxos de Exceção

E3. Campo Data Alteração

Ao editar um registro o campo data alteração recebe a data corrente/atual.

UC010 – Manter Atividade

Controle do Documento

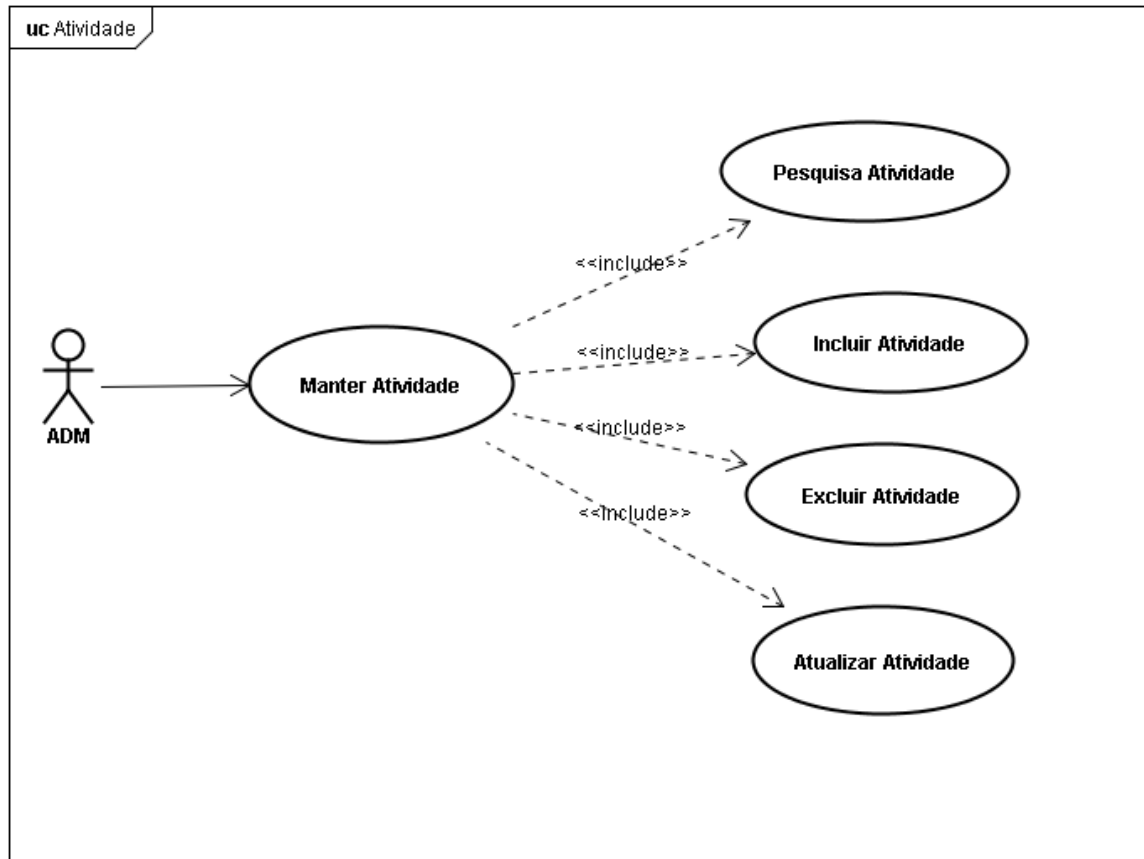
Versão	Autor	Data	Descrição
1.0	Maicon Fang	12/08/2013	Elaboração
1.1	Maicon Fang	16/08/2013	Correções
2.0	Maicon Fang	13/12/2013	Correções

Descrição

Este caso de uso serve para fazer o cadastramento de uma atividade.

Diagrama de Caso de Uso

DCU – Atividade.



Pré-condições

Este caso de uso pode iniciar somente se:

O sistema tiver executado o UC011 - Login

O usuário possuir permissão para cadastrar uma atividade.

Pós-condições

Após o fim normal deste caso de uso o sistema deve:
Ter salvo os dados da atividade.

Ator Primário

Administrador

Fluxo de Eventos Principal

- O sistema apresenta a tela (DV1)
- O sistema habilita o botão Cadastrar.
- O sistema habilita o botão Voltar.
- O sistema desabilita para edição o campo Data Criação.
- O sistema desabilita para edição o campo Data Alteração.
- O sistema desabilita para edição o campo Código.
- O sistema carrega a data corrente no campo Data Criação (R17).
- O sistema carrega a data corrente no campo Data Alteração (E3).
- O usuário preenche o campo nome.
- O usuário pressiona o botão Salvar. (A1).
- O caso de uso é finalizado.

Fluxos Alternativos

A1: Botão Voltar pressionado.

O usuário pressiona o botão voltar.

O sistema retorna a tela anterior.

Fluxos de Exceção

E3. Campo Data Alteração

Ao editar um registro o campo data alteração recebe a data corrente/atual.

UC011 – Manter Login

Controle do Documento

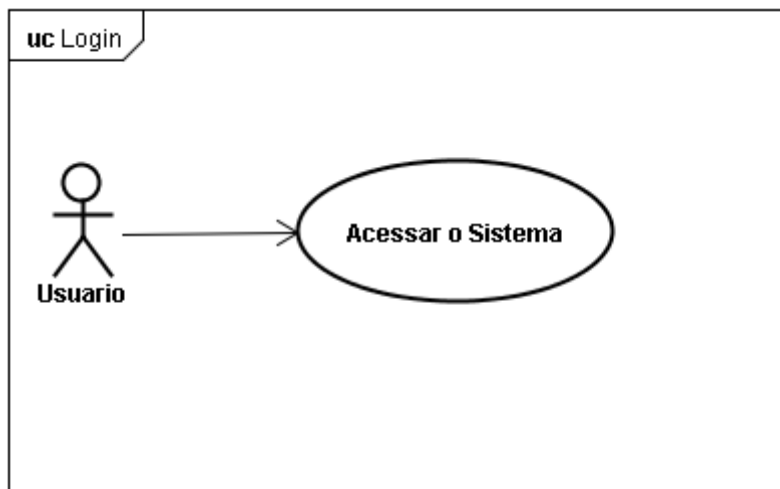
Versão	Autor	Data	Descrição
1.0	Maicon Fang	12/08/2013	Elaboração
1.1	Maicon Fang	16/08/2013	Correções
2.0	Maicon Fang	13/12/2013	Correções

Descrição

Este caso de uso serve para acessar o sistema.

Diagrama de Caso de Uso

DCU – Perfil.



Pré-condições

Este caso de uso pode iniciar somente se:

O usuário possuir permissão para acessar o sistema.

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

Ter permitido o acesso ao sistema.

Ator Primário

Administrador

Fluxo de Eventos Principal

O sistema apresenta a tela (DV1).
O sistema habilita o botão login.
O usuário preenche o campo login.
O usuário preenche o campo senha.
O usuário pressiona o botão Login. (E1).
O caso de uso é finalizado.

Fluxos Alternativos

A1: Botão Voltar pressionado.
O usuário pressiona o botão voltar.
O sistema retorna a tela anterior.

Fluxos de Exceção

E1. Login e senha incorretos.
O sistema exibe a mensagem “Não foi possível se autenticar. Login ou senha inválidos!”

UC500 – Exibir empresa/s na aplicação Android.

Controle do Documento

Versão	Autor	Data	Descrição
1.0	Maicon Fang	12/08/2013	Elaboração
1.1	Maicon Fang	16/08/2013	Correções

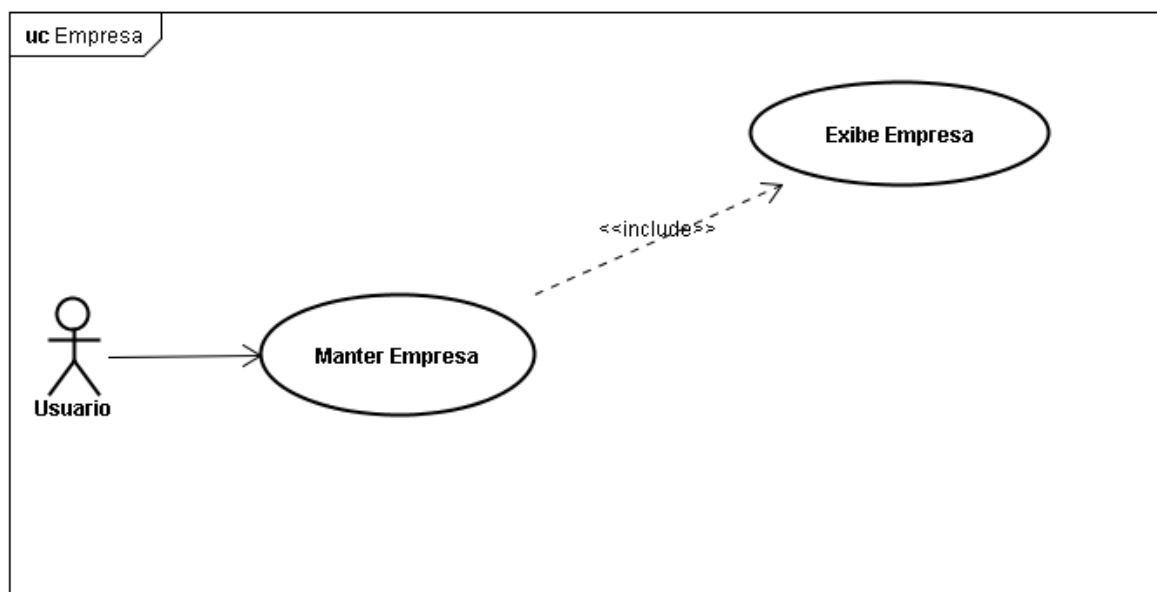
2.0	Maicon Fang	12/12/2013	Correções
2.1	Maicon Alexander Illes Fang	19/02/2014	Remoção do caso de uso 502

Descrição

Este caso de uso serve para exibir a/s empresa/s na aplicação Android.

Diagrama de Caso de Uso

DCU – Empresa.



Pré-condições

Este caso de uso pode iniciar somente se:

Existir conexão com a internet.

A aplicação estiver previamente instalada no dispositivo móvel.

Ter acessado a aplicação.

Web Service estiver ativo.

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

Ter exibido a/s empresa/s.

Ator Primário

Usuário

Fluxo de Eventos Principal

Ao inicializar o aplicativo as empresas são carregadas (DV1), (R10).

O usuário seleciona a empresa (A1), (A2).

O sistema chama o Use Case 501

O caso de uso é finalizado.

Fluxos Alternativos

A1: Botão “Voltar” pressionado.

O sistema retorna para tela anterior.

O caso de uso é finalizado

A2: Botão “Home” pressionado.

O sistema retorna para tela principal do dispositivo móvel.

O caso de uso é finalizado

Fluxos de Exceção

Não aplicável.

UC501 – Exibir produto/s na aplicação Android.

Controle do Documento

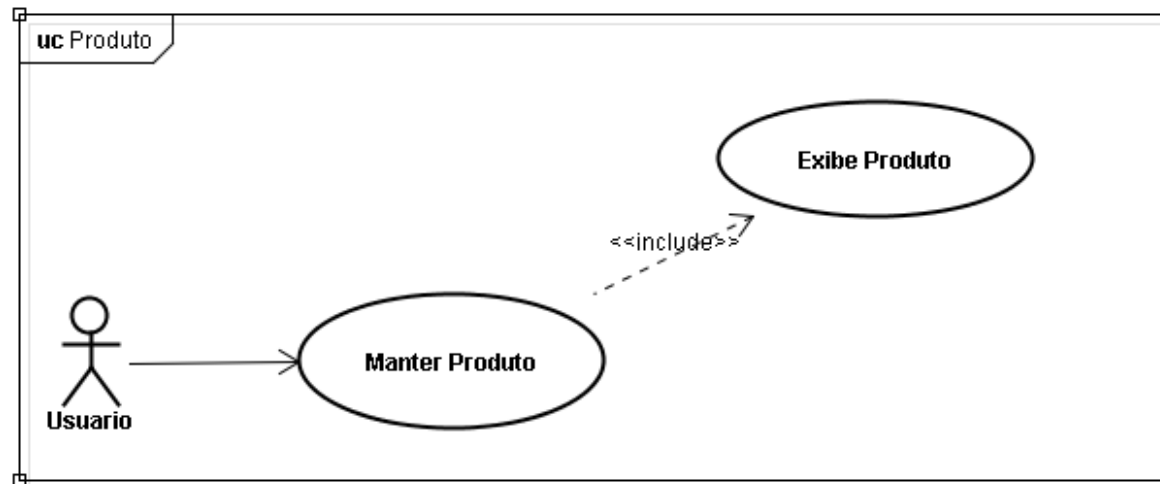
Versão	Autor	Data	Descrição
1.0	Maicon Fang	12/08/2013	Elaboração
1.1	Maicon Fang	16/08/2013	Correções
2.0	Maicon Fang	12/12/2013	Correções
2.1	Maicon Alexander Illes Fang	19/02/2014	Correções

Descrição

Este caso de uso serve para exibir os produtos na aplicação Android.

Diagrama de Caso de Uso

DCU – Produto.



Pré-condições

Este caso de uso pode iniciar somente se:

A aplicação estiver previamente instalada no dispositivo móvel.

Ter acessado á aplicação.

Ter executado o Use Case 500.

Web Service estiver ativo

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

Ter exibido os produtos.

Ator Primário

Usuário

Fluxo de Eventos Principal

Ao clicar na empresa o/s produto/s são exibidos (DV1), (R4), (R11), (E1), (E2)

O usuário seleciona o produto.

O sistema chama o Use case 502

O caso de uso é finalizado.

Fluxos Alternativos

A1: Botão “Voltar” pressionado.

O sistema retorna para tela anterior.

O sistema apresenta a tela com as empresas cadastradas.

O caso de uso é finalizado.

A2: Botão “Home” pressionado.

O sistema retorna para tela principal do dispositivo móvel.

O caso de uso é finalizado.

Fluxos de Exceção

Não aplicável.

UC502 – Exibir detalhes do produto na aplicação Android.

Controle do Documento

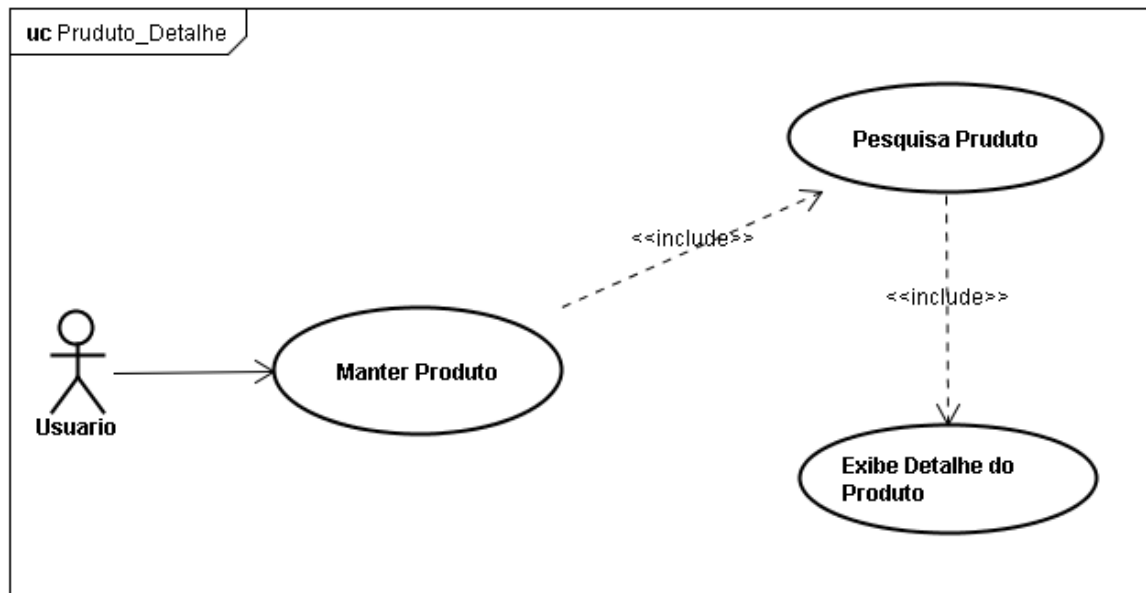
Versão	Autor	Data	Descrição
1.0	Maicon Fang	12/08/2013	Elaboração
1.1	Maicon Fang	16/08/2013	Correções
2.0	Maicon Fang	12/12/2013	Correções
2.1	Maicon Alexander Illes Fang	19/02/2014	Correções

Descrição

Este caso de uso serve para exibir um único produto na aplicação Android.

Diagrama de Caso de Uso

DCU – Detalhes do Produto.



Pré-condições

Este caso de uso pode iniciar somente se:

Existir conexão com a internet.

A aplicação estiver previamente instalada no dispositivo móvel.

Ter acessado a aplicação.

Ter executado o Use Case 501.

O Web Service estiver ativo.

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

Ter exibido o produto.

Ator Primário

Usuário

Fluxo de Eventos Principal

Á aplicação exibe o produto selecionado (E1), (E2)

O caso de uso é finalizado

Fluxos Alternativos

A1: Botão “Voltar” pressionado.

O sistema retorna para tela anterior.

O caso de uso é finalizado

A2: Botão “Home” pressionado.

O sistema retorna para tela principal do dispositivo móvel.

O caso de uso é finalizado

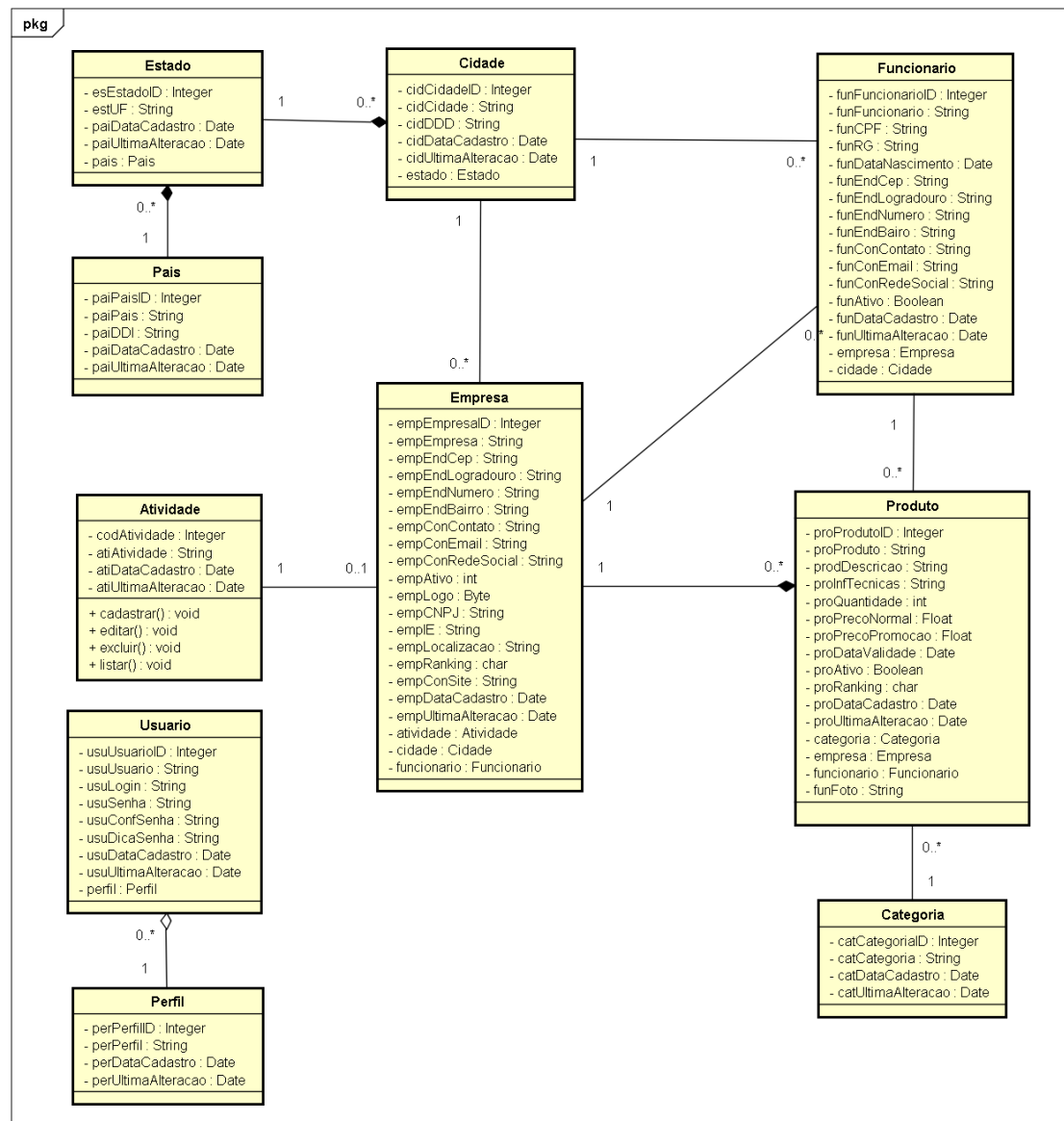
Fluxos de Exceção

Não aplicável.

APÊNDICE I - 9 - Template Modelo de Objetos

Diagrama de Classes com Atributos

Versão 2.1



APÊNDICE J - 10 - Template Casos de Uso

Diagrama de Caso de Uso com includes, extends, etc de todos os casos de uso.

Especificação Detalhada.

Versão 2.1

Histórico da Revisão

Data	Versão	Descrição	Autor
10/08/2013	1.0	Versão inicial	Maicon Alexander Illes Fang
11/12/2013	2.0	Versão secundaria	Maicon Alexander Illes Fang
24/02/2014	2.1	Ajustes e inserção de novos casos de usos.	Maicon Alexander Illes Fang

UC001 – Manter Empresa

Controle do Documento

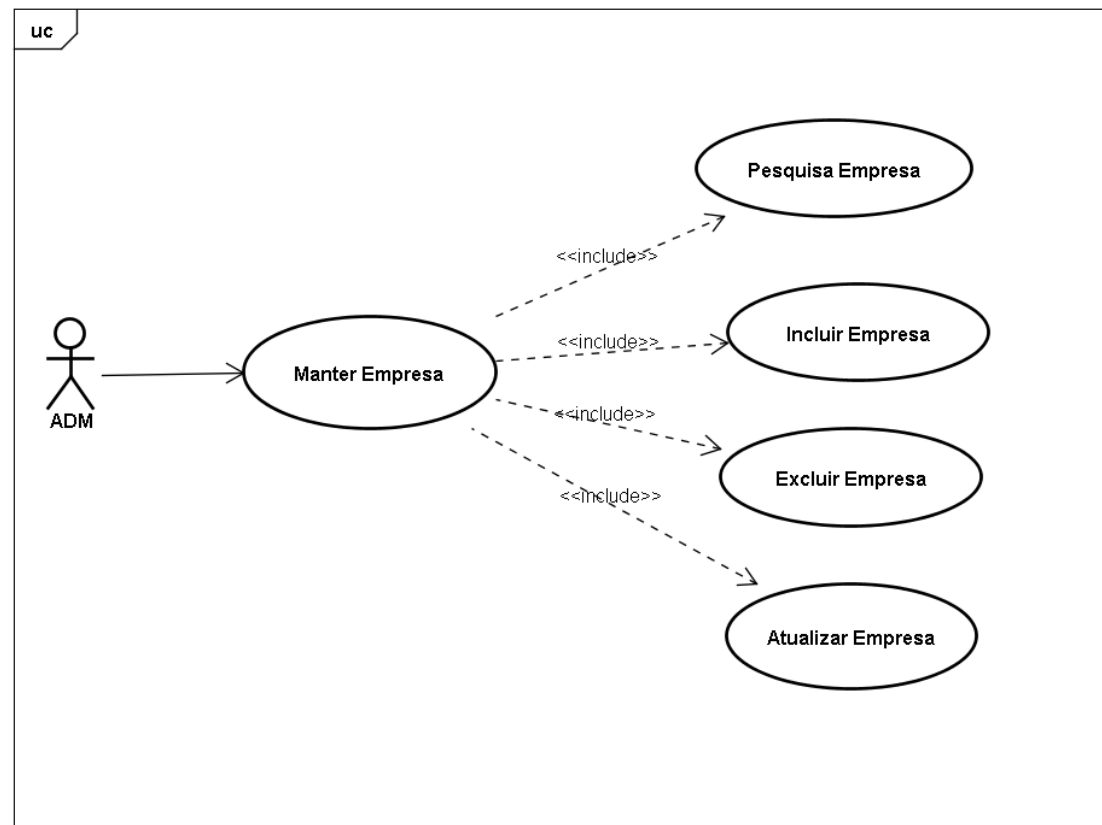
Versão	Autor	Data	Descrição
1.0	Maicon Fang	12/08/2013	Elaboração
1.1	Maicon Fang	16/08/2013	Correções
2.0	Maicon Fang	13/12/2013	Correções

Descrição

Este caso de uso serve para cadastrar uma empresa.

Diagrama de Caso de Uso

DCU - Empresa.



Pré-condições

Este caso de uso pode iniciar somente se:

- O sistema tiver executado o UC011 - Login
- O usuário possuir permissão para cadastrar uma empresa.
- O usuário tiver cadastrado uma cidade e atividade.

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

Ter salvo os dados da empresa.

Ator Primário

Administrador

Fluxo de Eventos Principal

- O sistema apresenta a tela (DV1)
- O sistema habilita o botão Cadastrar.
- O sistema carrega a data corrente no campo Data Criação (R17).
- O sistema carrega a data corrente no campo Data Alteração (E1).
- O sistema desabilita para edição o campo Data Criação.
- O sistema desabilita para edição o campo Data Alteração.
- O sistema desabilita para edição o campo Código.
- O sistema carrega a lista de Cidades cadastradas.

O sistema carrega a lista de Atividades cadastradas.

O usuário preenche o campo Nome.

O usuário seleciona o campo Cidade.

O usuário seleciona o campo Atividade.

O usuário seleciona o campo Foto.

O usuário preenche o campo CNPJ (R11).

O usuário preenche o campo IE.

O usuário preenche o campo CEP.

O usuário preenche o campo Logradouro.

O usuário preenche o campo Número.

O usuário preenche o campo Complemento.

O usuário preenche o campo Bairro.

O usuário preenche o campo Telefone.

O usuário preenche o campo Contato.

O usuário preenche o campo Site.

O usuário pressiona o botão Cadastrar. (A3)

O caso de uso é finalizado.

Fluxos Alternativos

A3: Botão Voltar pressionado.

O usuário pressiona o botão voltar.

O sistema retorna a tela anterior.

Fluxos de Exceção

E1. Campo Data Alteração.

Ao editar um registro o campo data alteração recebe a data corrente/atual.

UC002 – Manter Funcionário

Controle do Documento

Versão	Autor	Data	Descrição
1.0	Maicon Fang	12/08/2013	Elaboração
1.1	Maicon Fang	16/08/2013	Correções
2.0	Maicon Fang	13/12/2013	Correções

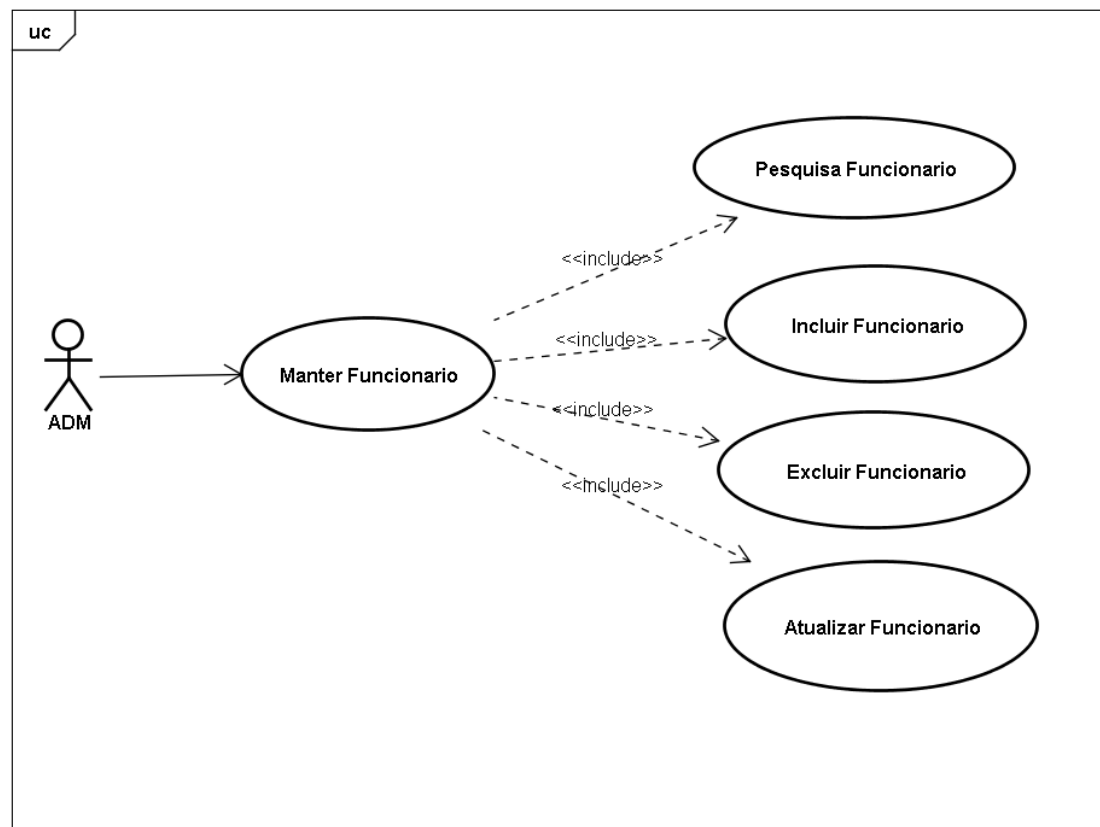
Especificação de Caso de Uso

Descrição

Este caso de uso serve para cadastrar um funcionário.

Diagrama de Caso de Uso

DCU – Funcionário.



Pré-condições

Este caso de uso pode iniciar somente se:

O sistema tiver executado o UC011 - Login

O usuário possuir permissão para cadastrar um funcionário.

O usuário tiver cadastrado uma cidade e empresa.

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

Ter salvo os dados do Funcionário.

Ator Primário

Administrador

Fluxo de Eventos Principal

O sistema apresenta a tela (DV1)

O sistema habilita o botão Cadastrar.

O sistema habilita o botão Voltar.

O sistema desabilita para edição o campo Data Criação (R17).

O sistema desabilita para edição o campo Data Alteração (E).

O sistema desabilita para edição o campo Código.

O sistema carrega a data corrente no campo Data Criação (A2).

O sistema carrega a data corrente no campo Data Alteração.

O sistema carrega uma lista de Empresas no combo Box Empresa.

O sistema carrega uma lista de Cidades no combo Box Cidade.

O usuário preenche o campo Nome.

O usuário seleciona o campo Cidade.

O usuário seleciona o campo Empresa.

O usuário seleciona o campo Foto.
O usuário preenche o campo CPF (R21).
O usuário preenche o campo RG.
O usuário preenche o campo Data Nascimento.
O usuário preenche o campo CEP.
O usuário preenche o campo Logradouro.
O usuário preenche o campo Bairro.
O usuário preenche o campo Número.
O usuário preenche o campo Complemento.
O usuário preenche o campo Contato.
O usuário preenche o campo E-mail.
O usuário preenche o campo Rede Social.
O usuário preenche o campo Ativo.
O usuário pressiona o botão Cadastrar. (A1)
O caso de uso é finalizado.

Fluxos Alternativos

A1: Botão Voltar pressionado.
O usuário pressiona o botão voltar.
O sistema retorna a tela anterior.

Fluxos de Exceção

E1. Campo Data Alteração.

Ao editar um registro o campo data alteração recebe a data corrente/atual.

UC003 – Manter Produto

Controle do Documento

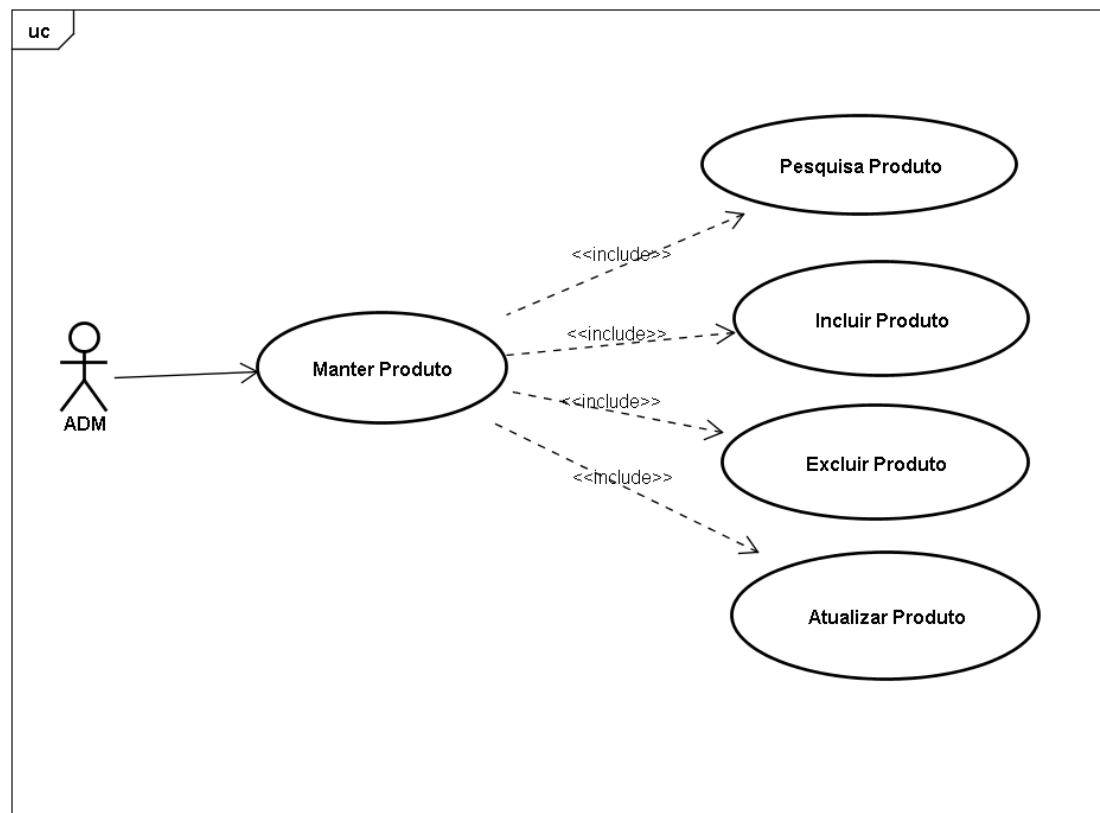
Versão	Autor	Data	Descrição
1.0	Maicon Fang	12/08/2013	Elaboração
1.1	Maicon Fang	16/08/2013	Correções
2.0	Maicon Fang	13/12/2013	Correções

Descrição

Este caso de uso serve para cadastrar um produto

Diagrama de Caso de Uso

DCU – Produto.



Pré-condições

Este caso de uso pode iniciar somente se:

O sistema tiver executado o UC011 - Login

O usuário possuir permissão para cadastrar um Produto.

O sistema deve ter um Funcionário, Empresa e Categoria cadastrada.

Pós-condições

Após o fim normal deste caso de uso o sistema deve:
Ter salvo os dados do Produto.

Ator Primário

Funcionário.

Fluxo de Eventos Principal

- O sistema apresenta a tela (DV1)
- O sistema habilita o botão Cadastrar.
- O sistema habilita o botão Voltar.
- O sistema desabilita para edição o campo Data Criação.
- O sistema desabilita para edição o campo Data Alteração.
- O sistema desabilita para edição o campo Código.
- O sistema carrega a data corrente no campo Data Criação (R17).
- O sistema carrega a data corrente no campo Data Alteração (E3).
- O sistema carrega o campo ativo preenchido e desabilitado (E2).
- O sistema carrega o combo Box Categoria.
- O sistema carrega o combo Box Empresa.
- O sistema carrega o combo Box Funcionário.
- O usuário seleciona o campo Foto do Produto.

O usuário seleciona o campo Categoria do Produto.
O usuário preenche o campo Nome.
O usuário preenche o campo Descrição.
O usuário preenche o campo Especificações Técnicas.
O usuário preenche o campo Quantidade.
O usuário preenche o campo Preço Normal.
O usuário preenche o campo Preço Promoção (R19).
O usuário preenche o campo Data Validade (R20).
O usuário pressiona o botão Salvar. (A1).
O caso de uso é finalizado.

Fluxos Alternativos

A1: Botão Voltar pressionado.
O usuário pressiona o botão voltar.
O sistema retorna a tela anterior.

Fluxos de Exceção

E1. Lista sem produto cadastrada
O sistema exibe a lista sem produtos cadastrados
O sistema desabilita os botões Editar, Excluir

E2. Campo ativo editar registro

O sistema habilita o campo com as opções ativo e inativo

E3. Campo Data Alteração

Ao editar um registro o campo data alteração recebe a data corrente/atual.

UC004 – Manter Categoria

Controle do Documento

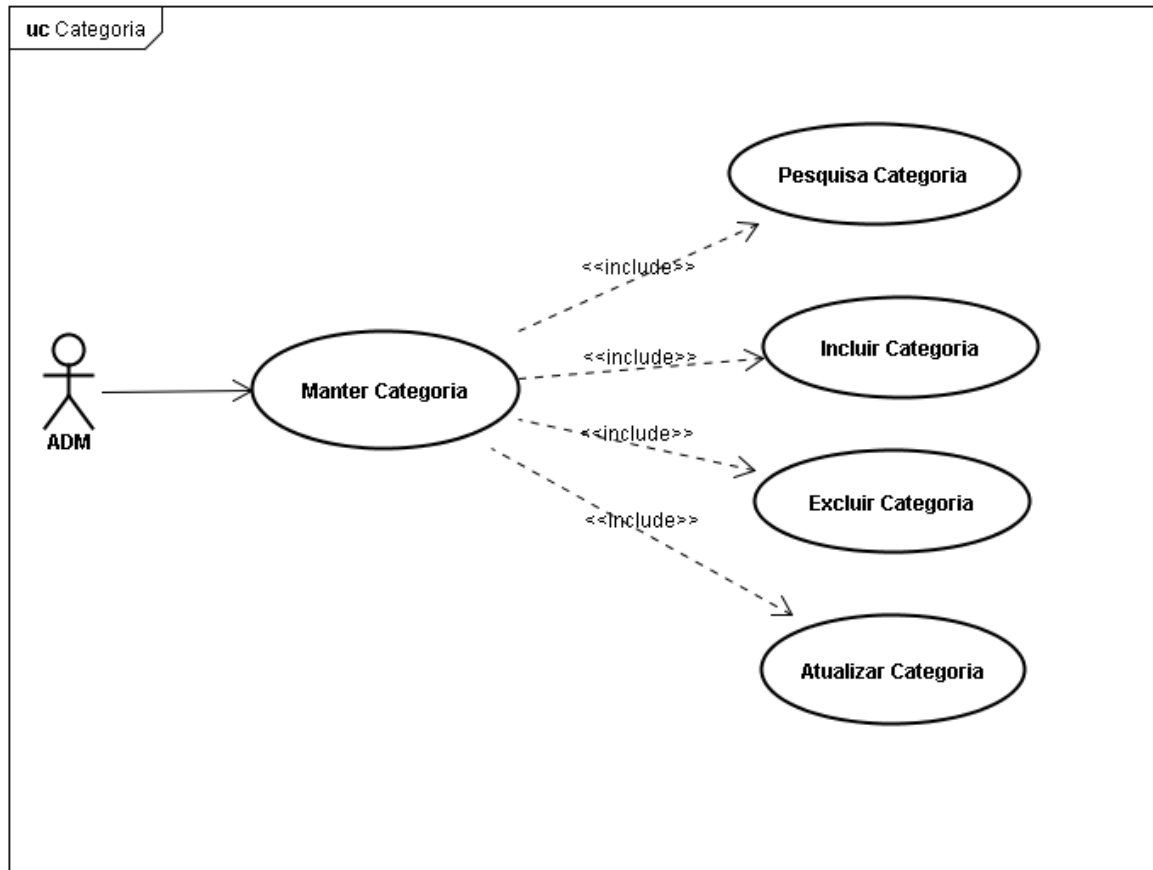
Versão	Autor	Data	Descrição
1.0	Maicon Fang	12/08/2013	Elaboração
1.1	Maicon Fang	16/08/2013	Correções
2.0	Maicon Fang	13/12/2013	Correções

Descrição

Este caso de uso serve para fazer o cadastramento de uma categoria.

Diagrama de Caso de Uso

DCU – Categoria.



Pré-condições

Este caso de uso pode iniciar somente se:

O sistema tiver executado o UC011 - Login

O usuário possuir permissão para cadastrar uma categoria.

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

Ter salvo os dados da categoria.

Ator Primário

Administrador

Fluxo de Eventos Principal

O sistema apresenta a tela (DV1)

O sistema habilita o botão Cadastrar.

O sistema habilita o botão Voltar.

O sistema desabilita para edição o campo Data Criação.

O sistema desabilita para edição o campo Data Alteração.

O sistema desabilita para edição o campo Código.

O sistema carrega a data corrente no campo Data Criação (R17).

O sistema carrega a data corrente no campo Data Alteração (E3).

O usuário preenche o campo nome.

O usuário pressiona o botão Salvar. (A1).

O caso de uso é finalizado.

Fluxos Alternativos

A1: Botão Voltar pressionado.

O usuário pressiona o botão voltar.

O sistema retorna a tela anterior.

Fluxos de Exceção

E3. Campo Data Alteração

Ao editar um registro o campo data alteração recebe a data corrente/atual.

UC005 – Manter Pais

Controle do Documento

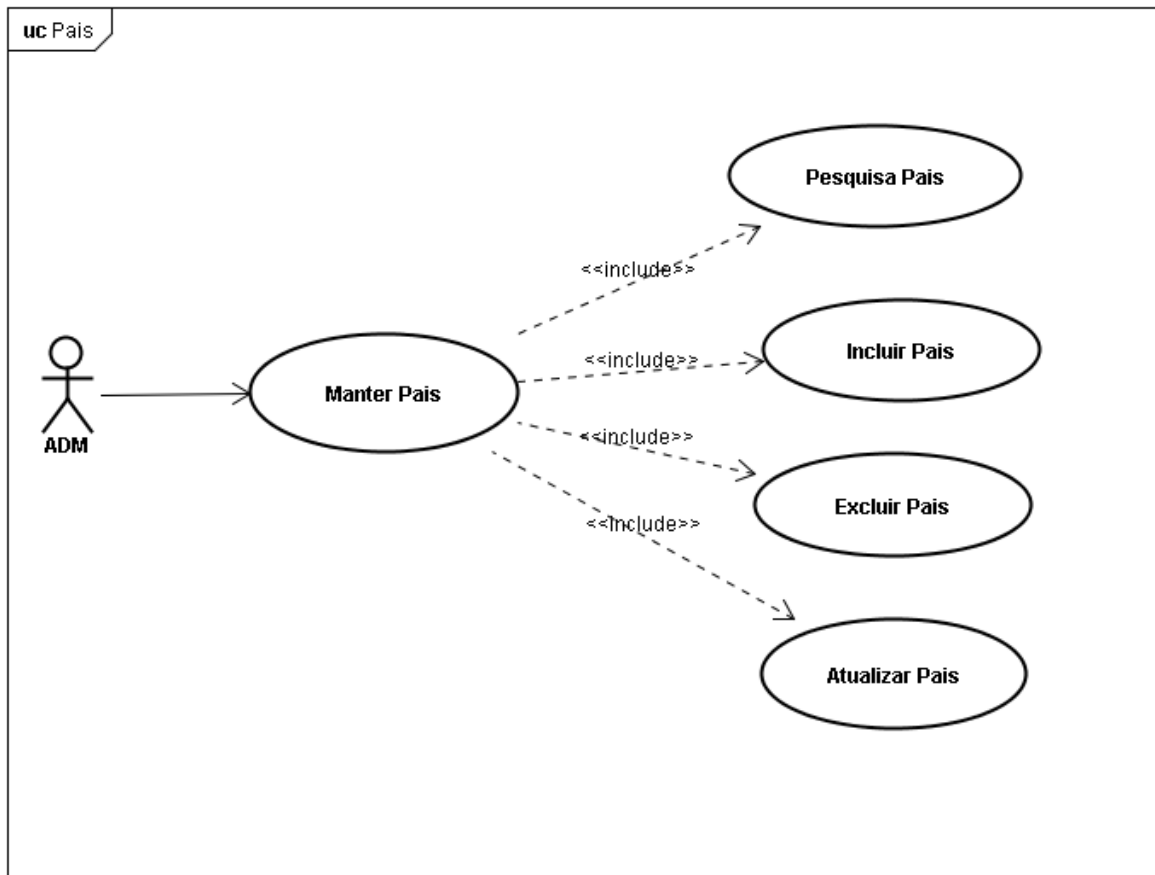
Versão	Autor	Data	Descrição
1.0	Maicon Fang	12/08/2013	Elaboração
1.1	Maicon Fang	16/08/2013	Correções
2.0	Maicon Fang	13/12/2013	Correções

Descrição

Este caso de uso serve para fazer o cadastramento de um pais.

Diagrama de Caso de Uso

DCU – Pais.



Pré-condições

Este caso de uso pode iniciar somente se:

O sistema tiver executado o UC011 - Login

O usuário possuir permissão para cadastrar um país.

Pós-condições

Após o fim normal deste caso de uso o sistema deve:
Ter salvo os dados do país.

Ator Primário

Administrador

Fluxo de Eventos Principal

- O sistema apresenta a tela (DV1)
- O sistema habilita o botão Cadastrar.
- O sistema habilita o botão Voltar.
- O sistema desabilita para edição o campo Data Criação.
- O sistema desabilita para edição o campo Data Alteração.
- O sistema desabilita para edição o campo Código.
- O sistema carrega a data corrente no campo Data Criação (R17).
- O sistema carrega a data corrente no campo Data Alteração (E3).
- O usuário preenche o campo nome.
- O usuário preenche o campo DDI.
- O usuário pressiona o botão Salvar. (A1).
- O caso de uso é finalizado.

Fluxos Alternativos

A1: Botão Voltar pressionado.

O usuário pressiona o botão voltar.

O sistema retorna a tela anterior.

Fluxos de Exceção

E3. Campo Data Alteração

Ao editar um registro o campo data alteração recebe a data corrente/atual.

UC006 – Manter Estado

Controle do Documento

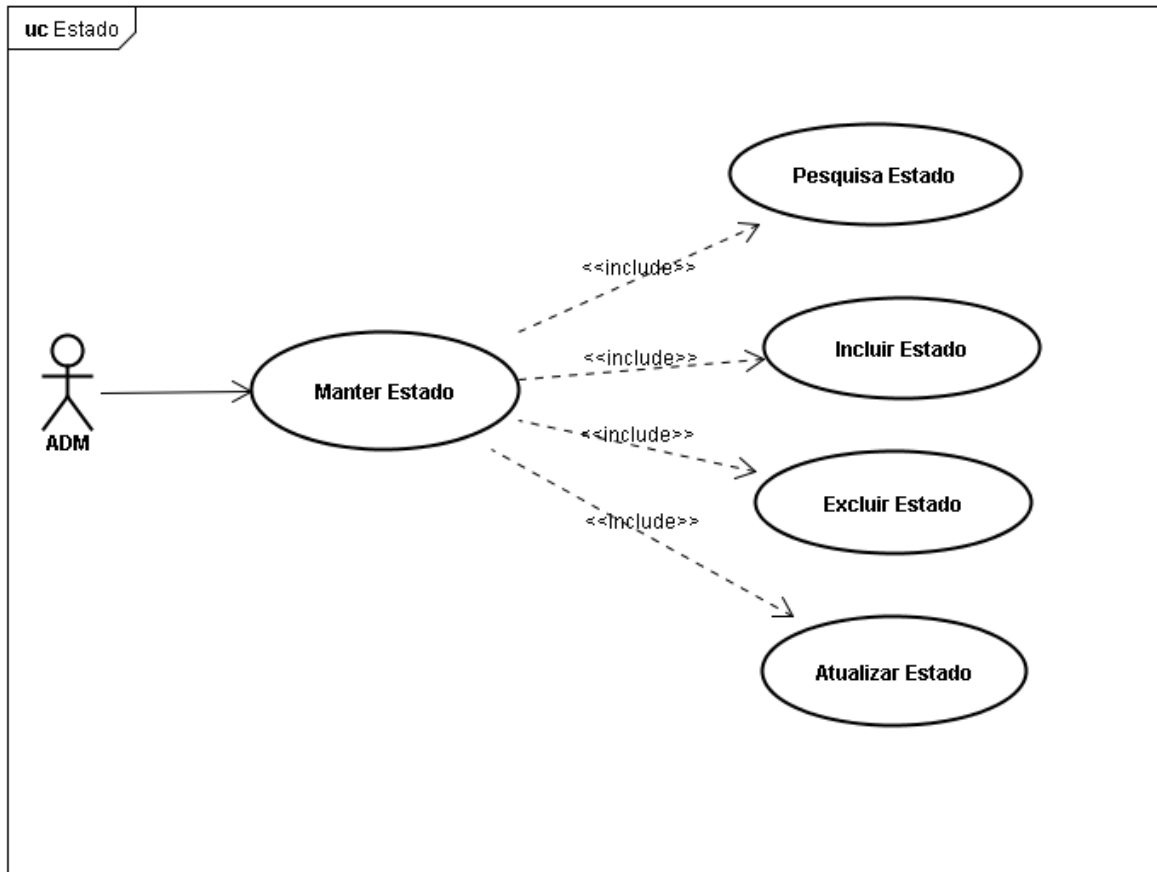
Versão	Autor	Data	Descrição
1.0	Maicon Fang	12/08/2013	Elaboração
1.1	Maicon Fang	16/08/2013	Correções
2.0	Maicon Fang	13/12/2013	Correções

Descrição

Este caso de uso serve para fazer o cadastramento de um estado.

Diagrama de Caso de Uso

DCU – Estado.



Pré-condições

Este caso de uso pode iniciar somente se:

O sistema tiver executado o UC011 - Login

O usuário possuir permissão para cadastrar um estado.

Pós-condições

Após o fim normal deste caso de uso o sistema deve:
Ter salvo os dados do estado.

Ator Primário

Administrador

Fluxo de Eventos Principal

- O sistema apresenta a tela (DV1)
- O sistema habilita o botão Cadastrar.
- O sistema habilita o botão Voltar.
- O sistema carrega os países no combobox pais.
- O sistema desabilita para edição o campo Data Criação.
- O sistema desabilita para edição o campo Data Alteração.
- O sistema desabilita para edição o campo Código.
- O sistema carrega a data corrente no campo Data Criação (R17).
- O sistema carrega a data corrente no campo Data Alteração (E3).
- O usuário preenche o campo nome.
- O usuário preenche o campo UF.
- O usuário seleciona o pais no combobox o pais.
- O usuário pressiona o botão Salvar. (A1).
- O caso de uso é finalizado.

Fluxos Alternativos

A1: Botão Voltar pressionado.

O usuário pressiona o botão voltar.

O sistema retorna a tela anterior.

Fluxos de Exceção

E3. Campo Data Alteração

Ao editar um registro o campo data alteração recebe a data corrente/atual.

UC007 – Manter Cidade

Controle do Documento

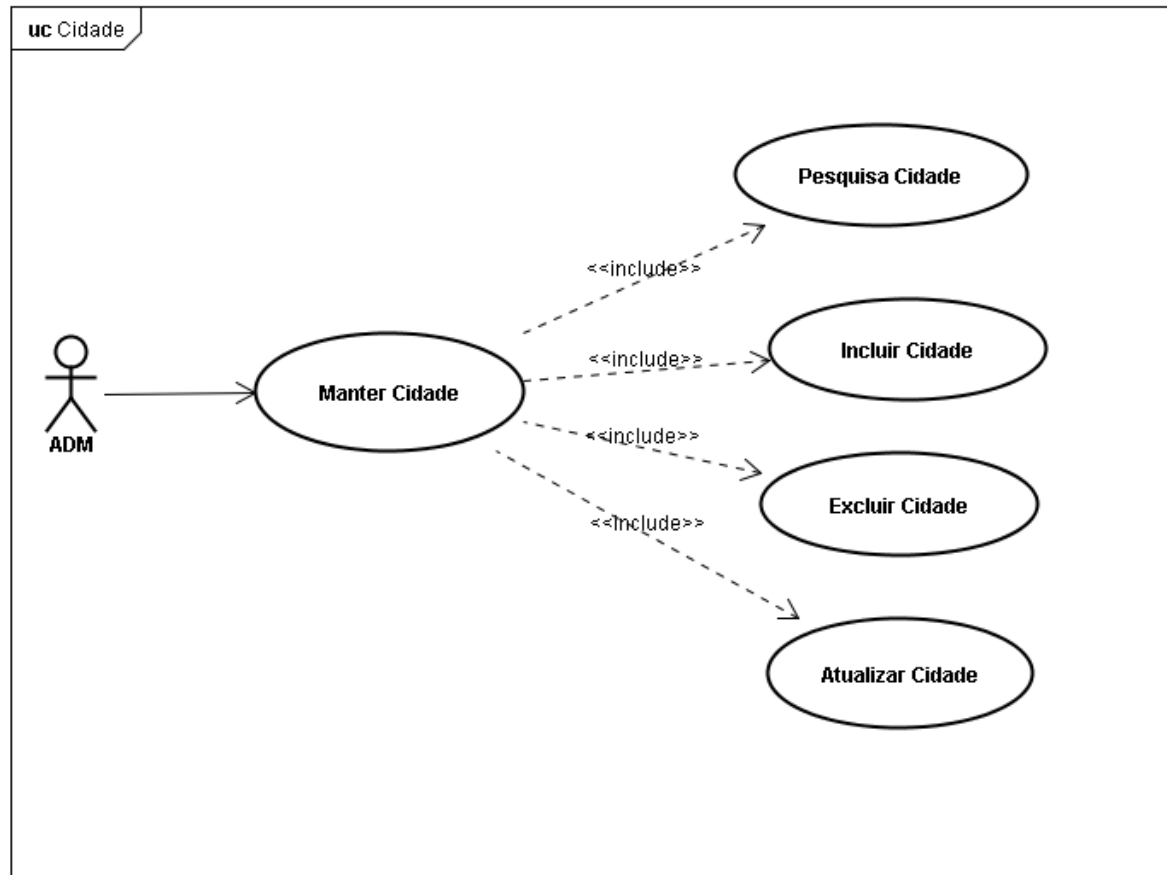
Versão	Autor	Data	Descrição
1.0	Maicon Fang	12/08/2013	Elaboração
1.1	Maicon Fang	16/08/2013	Correções
2.0	Maicon Fang	13/12/2013	Correções

Descrição

Este caso de uso serve para fazer o cadastramento de uma cidade.

Diagrama de Caso de Uso

DCU – Cidade.



Pré-condições

Este caso de uso pode iniciar somente se:

O sistema tiver executado o UC011 - Login

O usuário possuir permissão para cadastrar uma cidade.

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

Ter salvo os dados da cidade.

Ator Primário

Administrador

Fluxo de Eventos Principal

O sistema apresenta a tela (DV1)

O sistema habilita o botão Cadastrar.

O sistema habilita o botão Voltar.

O sistema carrega os estados no combobox estado.

O sistema desabilita para edição o campo Data Criação.

O sistema desabilita para edição o campo Data Alteração.

O sistema desabilita para edição o campo Código.

O sistema carrega a data corrente no campo Data Criação (R17).

O sistema carrega a data corrente no campo Data Alteração (E3).

O usuário preenche o campo nome.

O usuário preenche o campo DDD.

O usuário seleciona o estado no combobox o estado.

O usuário pressiona o botão Salvar. (A1).

O caso de uso é finalizado.

Fluxos Alternativos

A1: Botão Voltar pressionado.

O usuário pressiona o botão voltar.

O sistema retorna a tela anterior.

Fluxos de Exceção

E3. Campo Data Alteração

Ao editar um registro o campo data alteração recebe a data corrente/atual.

UC008 – Manter Usuário

Controle do Documento

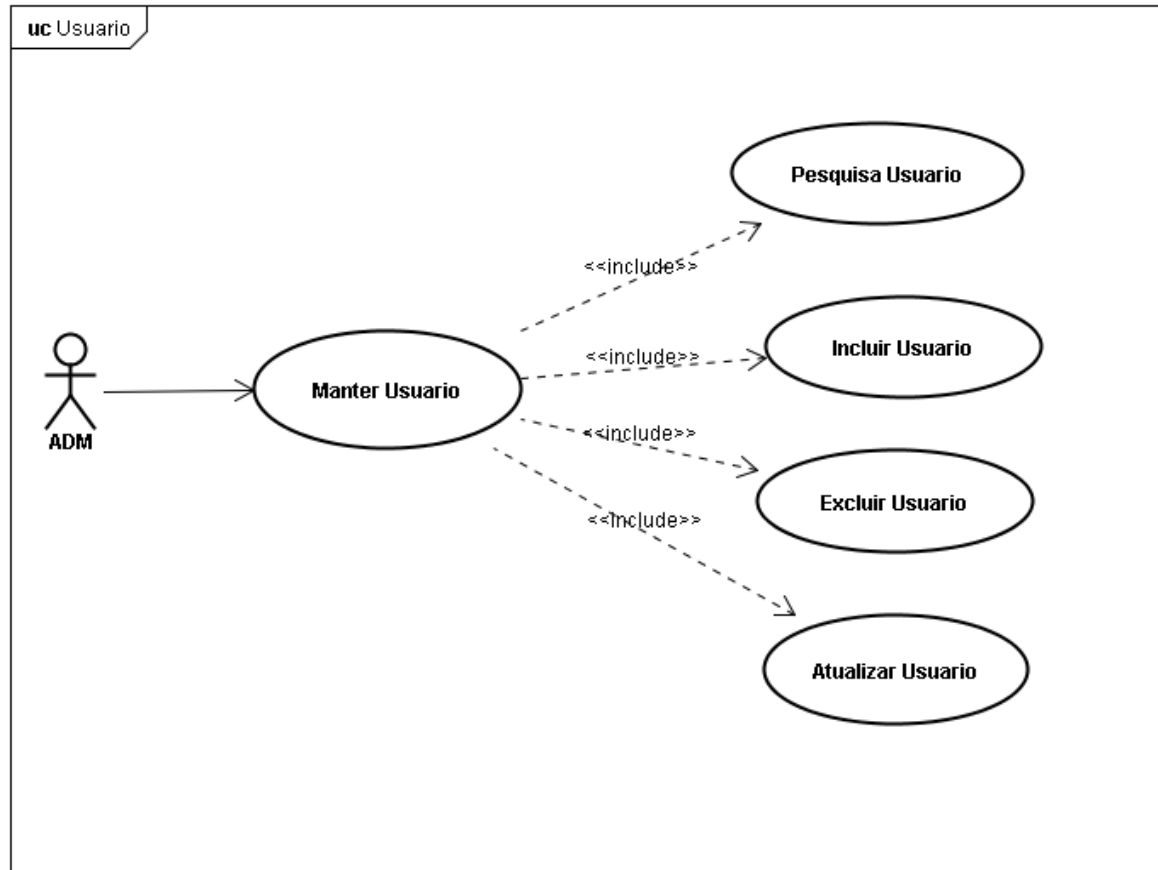
Versão	Autor	Data	Descrição
1.0	Maicon Fang	12/08/2013	Elaboração
1.1	Maicon Fang	16/08/2013	Correções
2.0	Maicon Fang	13/12/2013	Correções

Descrição

Este caso de uso serve para fazer o cadastramento de um usuário.

Diagrama de Caso de Uso

DCU – Usuário.



Pré-condições

Este caso de uso pode iniciar somente se:

O sistema tiver executado o UC011 - Login

O usuário possuir permissão para cadastrar um usuário.

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

Ter salvo os dados do usuário.

Ator Primário

Administrador

Fluxo de Eventos Principal

O sistema apresenta a tela (DV1)

O sistema habilita o botão Cadastrar.

O sistema habilita o botão Voltar.

O sistema carrega os perfis no combobox perfil.

O usuário preenche o campo matrícula.

O usuário preenche o campo nome.

O usuário preenche o login.

O usuário preenche o campo e-mail.

O usuário preenche o campo senha.

O usuário preenche o campo confirmar senha.

O usuário seleciona o perfil no combobox perfil.

O usuário pressiona o botão Salvar. (A1).

O caso de uso é finalizado.

Fluxos Alternativos

A1: Botão Voltar pressionado.

O usuário pressiona o botão voltar.

O sistema retorna a tela anterior.

Fluxos de Exceção

N/A

UC009 – Manter Perfil

Controle do Documento

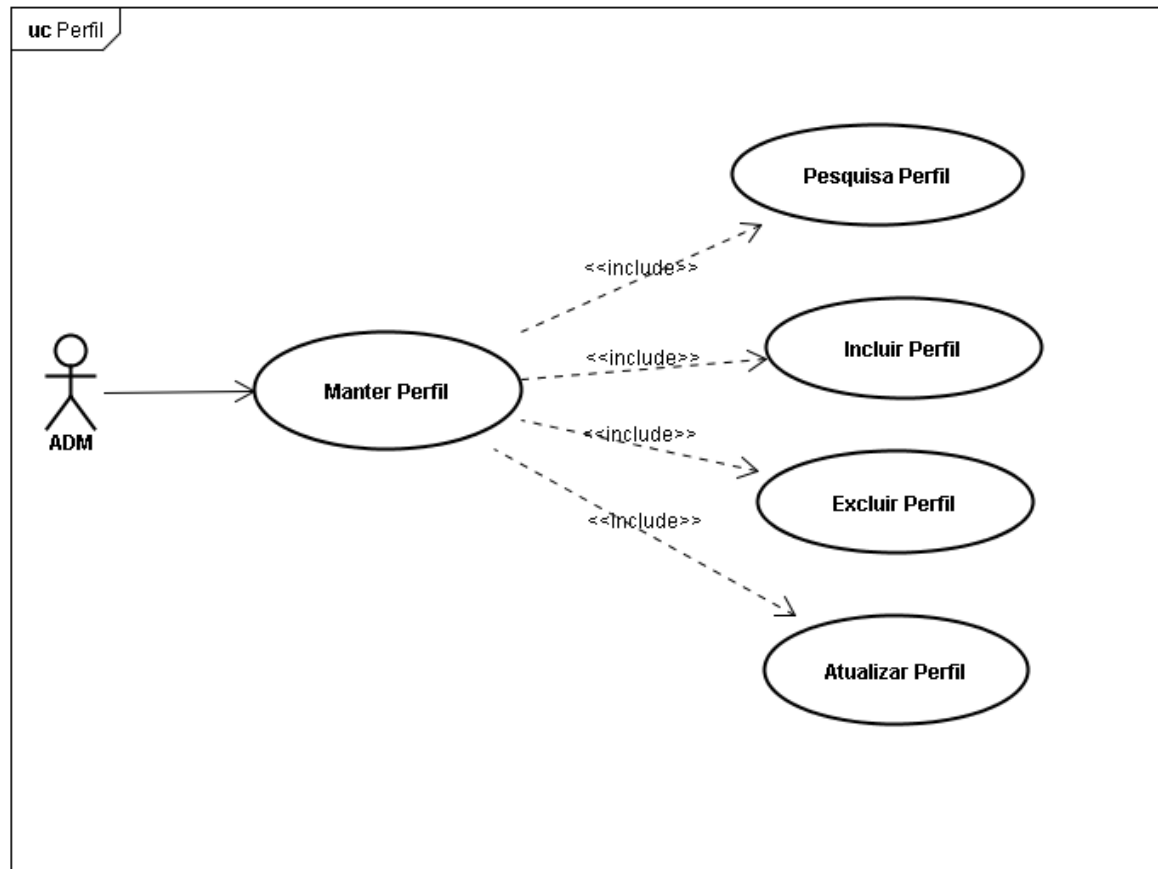
Versão	Autor	Data	Descrição
1.0	Maicon Fang	12/08/2013	Elaboração
1.1	Maicon Fang	16/08/2013	Correções
2.0	Maicon Fang	13/12/2013	Correções

Descrição

Este caso de uso serve para fazer o cadastramento de um perfil.

Diagrama de Caso de Uso

DCU – Perfil.



Pré-condições

Este caso de uso pode iniciar somente se:

O sistema tiver executado o UC011 - Login

O usuário possuir permissão para cadastrar um perfil.

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

Ter salvo os dados do perfil.

Ator Primário

Administrador

Fluxo de Eventos Principal

O sistema apresenta a tela (DV1)

O sistema habilita o botão Cadastrar.

O sistema habilita o botão Voltar.

O sistema desabilita para edição o campo Data Criação.

O sistema desabilita para edição o campo Data Alteração.

O sistema desabilita para edição o campo Código.

O sistema carrega a data corrente no campo Data Criação (R17).

O sistema carrega a data corrente no campo Data Alteração (E3).

O usuário preenche o campo nome.

O usuário pressiona o botão Salvar. (A1).

O caso de uso é finalizado.

Fluxos Alternativos

A1: Botão Voltar pressionado.

O usuário pressiona o botão voltar.

O sistema retorna a tela anterior.

Fluxos de Exceção

E3. Campo Data Alteração

Ao editar um registro o campo data alteração recebe a data corrente/atual.

UC010 – Manter Atividade

Controle do Documento

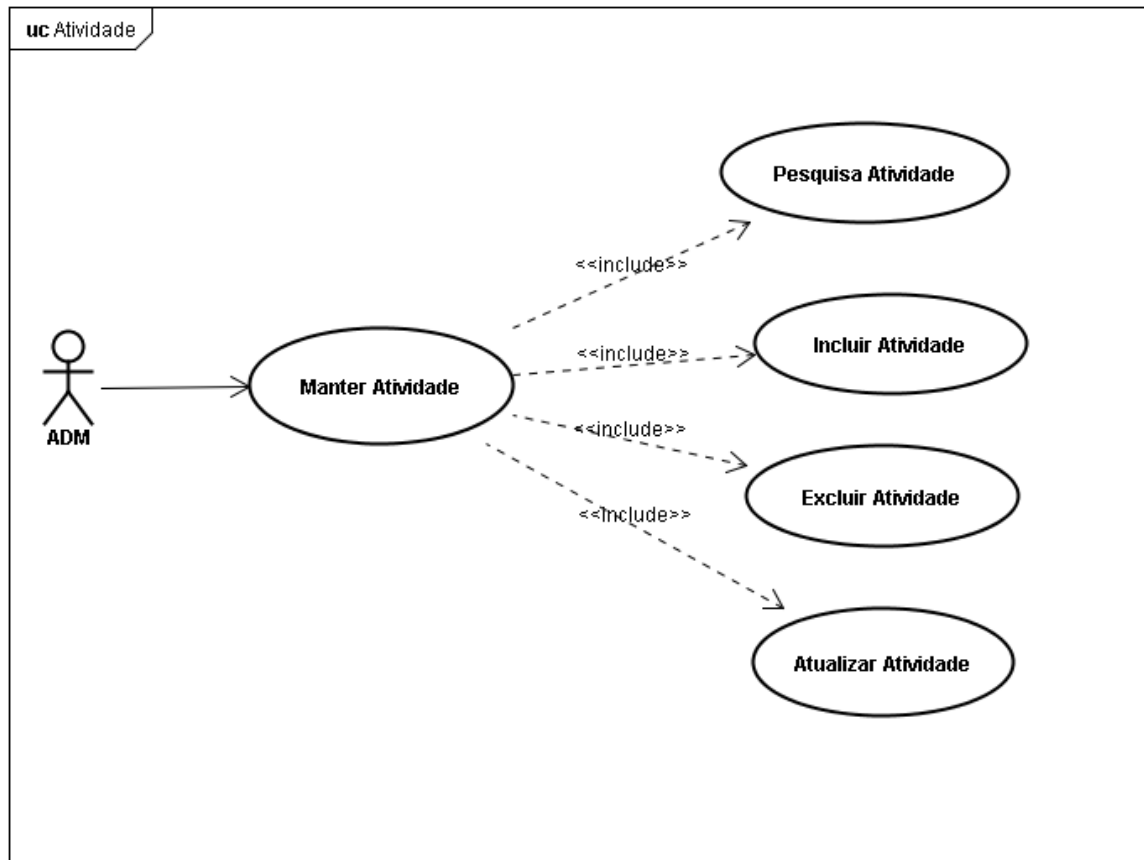
Versão	Autor	Data	Descrição
1.0	Maicon Fang	12/08/2013	Elaboração
1.1	Maicon Fang	16/08/2013	Correções
2.0	Maicon Fang	13/12/2013	Correções

Descrição

Este caso de uso serve para fazer o cadastramento de uma atividade.

Diagrama de Caso de Uso

DCU – Atividade.



Pré-condições

Este caso de uso pode iniciar somente se:

O sistema tiver executado o UC011 - Login

O usuário possuir permissão para cadastrar uma atividade.

Pós-condições

Após o fim normal deste caso de uso o sistema deve:
Ter salvo os dados da atividade.

Ator Primário

Administrador

Fluxo de Eventos Principal

- O sistema apresenta a tela (DV1)
- O sistema habilita o botão Cadastrar.
- O sistema habilita o botão Voltar.
- O sistema desabilita para edição o campo Data Criação.
- O sistema desabilita para edição o campo Data Alteração.
- O sistema desabilita para edição o campo Código.
- O sistema carrega a data corrente no campo Data Criação (R17).
- O sistema carrega a data corrente no campo Data Alteração (E3).
- O usuário preenche o campo nome.
- O usuário pressiona o botão Salvar. (A1).
- O caso de uso é finalizado.

Fluxos Alternativos

A1: Botão Voltar pressionado.

O usuário pressiona o botão voltar.

O sistema retorna a tela anterior.

Fluxos de Exceção

E3. Campo Data Alteração

Ao editar um registro o campo data alteração recebe a data corrente/atual.

UC011 – Manter Login

Controle do Documento

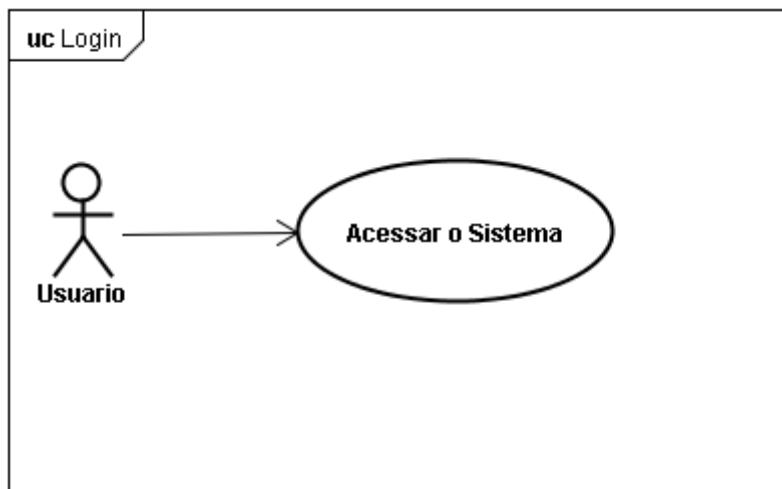
Versão	Autor	Data	Descrição
1.0	Maicon Fang	12/08/2013	Elaboração
1.1	Maicon Fang	16/08/2013	Correções
2.0	Maicon Fang	13/12/2013	Correções

Descrição

Este caso de uso serve para acessar o sistema.

Diagrama de Caso de Uso

DCU – Perfil.



Pré-condições

Este caso de uso pode iniciar somente se:

O usuário possuir permissão para acessar o sistema.

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

Ter permitido o acesso ao sistema.

Ator Primário

Administrador

Fluxo de Eventos Principal

O sistema apresenta a tela (DV1).
 O sistema habilita o botão login.
 O usuário preenche o campo login.
 O usuário preenche o campo senha.
 O usuário pressiona o botão Login. (E1).
 O caso de uso é finalizado.

Fluxos Alternativos

A1: Botão Voltar pressionado.
 O usuário pressiona o botão voltar.
 O sistema retorna a tela anterior.

Fluxos de Exceção

E1. Login e senha incorretos.
 O sistema exibe a mensagem “Não foi possível se autenticar. Login ou senha inválidos!”

UC500 – Exibir empresa/s na aplicação Android.

Controle do Documento

Versão	Autor	Data	Descrição
1.0	Maicon Fang	12/08/2013	Elaboração
1.1	Maicon Fang	16/08/2013	Correções

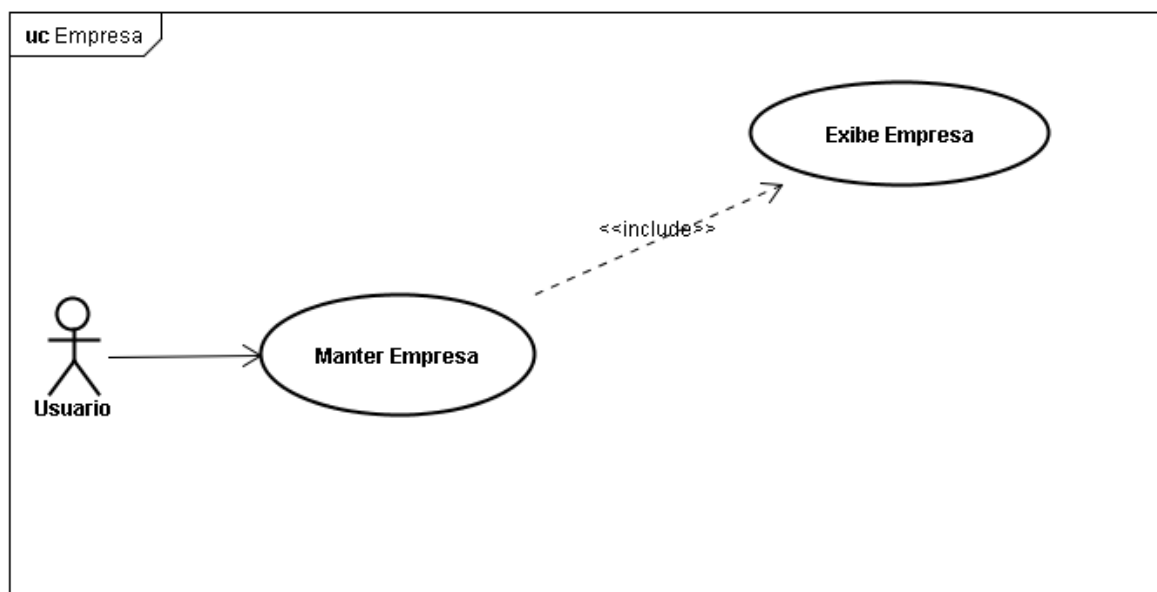
2.0	Maicon Fang	12/12/2013	Correções
2.1	Maicon Alexander Illes Fang	19/02/2014	Remoção do caso de uso 502

Descrição

Este caso de uso serve para exibir a/s empresa/s na aplicação Android.

Diagrama de Caso de Uso

DCU – Empresa.



Pré-condições

Este caso de uso pode iniciar somente se:

Existir conexão com a internet.

A aplicação estiver previamente instalada no dispositivo móvel.

Ter acessado a aplicação.

Web Service estiver ativo.

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

Ter exibido a/s empresa/s.

Ator Primário

Usuário

Fluxo de Eventos Principal

Ao inicializar o aplicativo as empresas são carregadas (DV1), (R10).

O usuário seleciona a empresa (A1), (A2).

O sistema chama o Use Case 501

O caso de uso é finalizado.

Fluxos Alternativos

A1: Botão “Voltar” pressionado.

O sistema retorna para tela anterior.

O caso de uso é finalizado

A2: Botão “Home” pressionado.

O sistema retorna para tela principal do dispositivo móvel.

O caso de uso é finalizado

Fluxos de Exceção

Não aplicável.

UC501 – Exibir produto/s na aplicação Android.

Controle do Documento

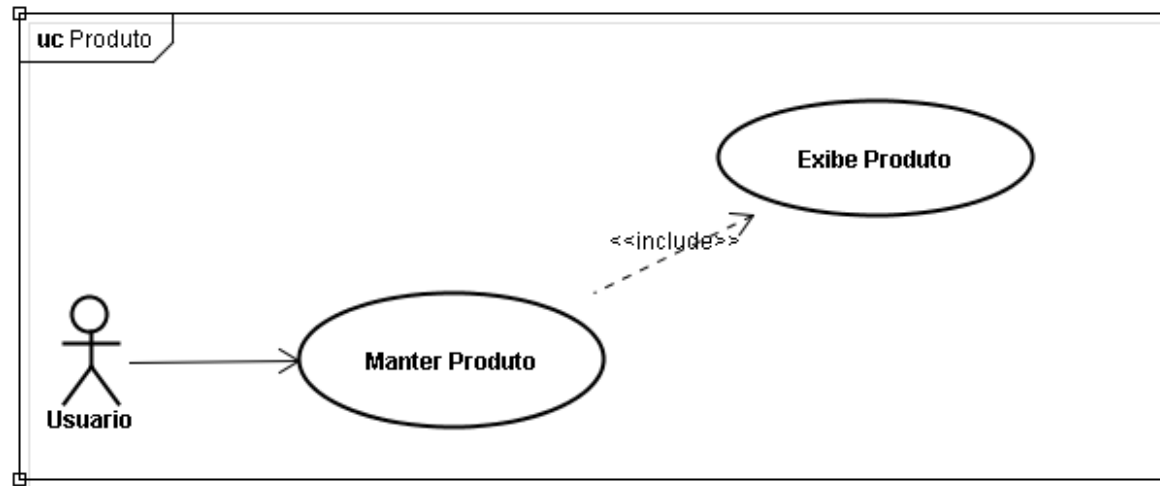
Versão	Autor	Data	Descrição
1.0	Maicon Fang	12/08/2013	Elaboração
1.1	Maicon Fang	16/08/2013	Correções
2.0	Maicon Fang	12/12/2013	Correções
2.1	Maicon Alexander Illes Fang	19/02/2014	Correções

Descrição

Este caso de uso serve para exibir os produtos na aplicação Android.

Diagrama de Caso de Uso

DCU – Produto.



Pré-condições

Este caso de uso pode iniciar somente se:

A aplicação estiver previamente instalada no dispositivo móvel.

Ter acessado á aplicação.

Ter executado o Use Case 500.

Web Service estiver ativo

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

Ter exibido os produtos.

Ator Primário

Usuário

Fluxo de Eventos Principal

Ao clicar na empresa o/s produto/s são exibidos (DV1), (R4), (R11), (E1), (E2)

O usuário seleciona o produto.

O sistema chama o Use case 502

O caso de uso é finalizado.

Fluxos Alternativos

A1: Botão “Voltar” pressionado.

O sistema retorna para tela anterior.

O sistema apresenta a tela com as empresas cadastradas.

O caso de uso é finalizado.

A2: Botão “Home” pressionado.

O sistema retorna para tela principal do dispositivo móvel.

O caso de uso é finalizado.

Fluxos de Exceção

Não aplicável.

UC502 – Exibir detalhes do produto na aplicação Android.

Controle do Documento

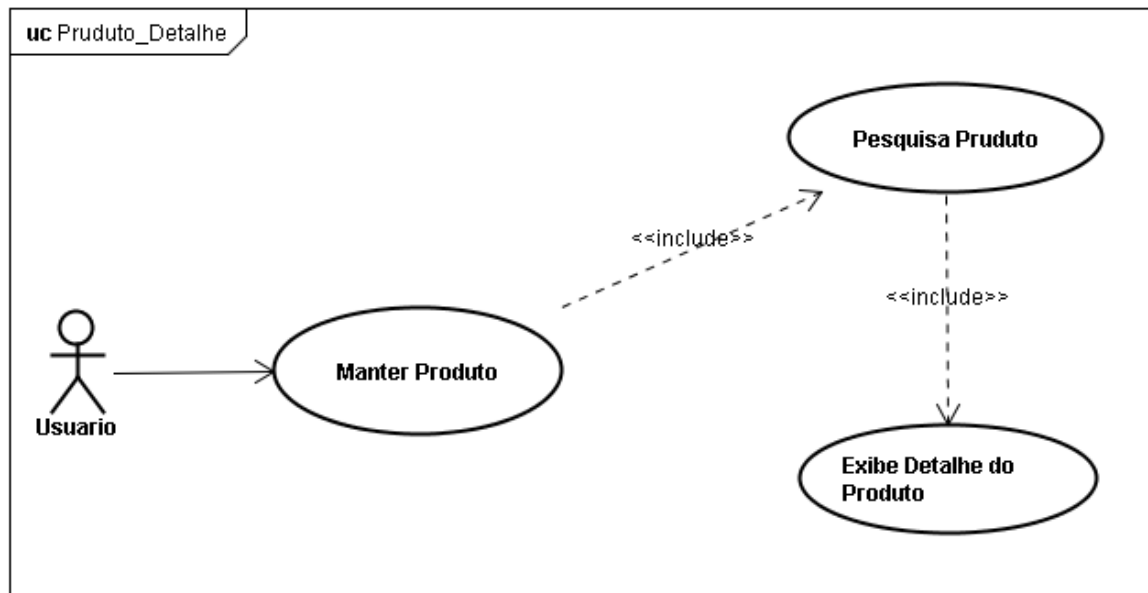
Versão	Autor	Data	Descrição
1.0	Maicon Fang	12/08/2013	Elaboração
1.1	Maicon Fang	16/08/2013	Correções
2.0	Maicon Fang	12/12/2013	Correções
2.1	Maicon Alexander Illes Fang	19/02/2014	Correções

Descrição

Este caso de uso serve para exibir um único produto na aplicação Android.

Diagrama de Caso de Uso

DCU – Detalhes do Produto.



Pré-condições

Este caso de uso pode iniciar somente se:

Existir conexão com a internet.

A aplicação estiver previamente instalada no dispositivo móvel.

Ter acessado a aplicação.

Ter executado o Use Case 501.

O Web Service estiver ativo.

Pós-condições

Após o fim normal deste caso de uso o sistema deve:
Ter exibido o produto.

Ator Primário

Usuário

Fluxo de Eventos Principal

Á aplicação exibe o produto selecionado (E1), (E2)

O caso de uso é finalizado

Fluxos Alternativos

A1: Botão “Voltar” pressionado.

O sistema retorna para tela anterior.

O caso de uso é finalizado

A2: Botão “Home” pressionado.

O sistema retorna para tela principal do dispositivo móvel.

O caso de uso é finalizado

Fluxos de Exceção

Não aplicável.

APÊNDICE K - 11 - Template Diagramas de Sequencia

Diagramas de sequência de todos os fluxos de todos os casos de uso.

Versão 2.1

Histórico da Revisão

Data	Versão	Descrição	Autor
10/08/2013	1.0	Versão inicial	Maicon Alexander Illes Fang
11/12/2013	2.0	Versão secundaria	Maicon Alexander Illes Fang
24/02/2014	2.1	Ajustes e inserção de novos casos de usos.	Maicon Alexander Illes Fang

Diagrama de Sequência – Empresa

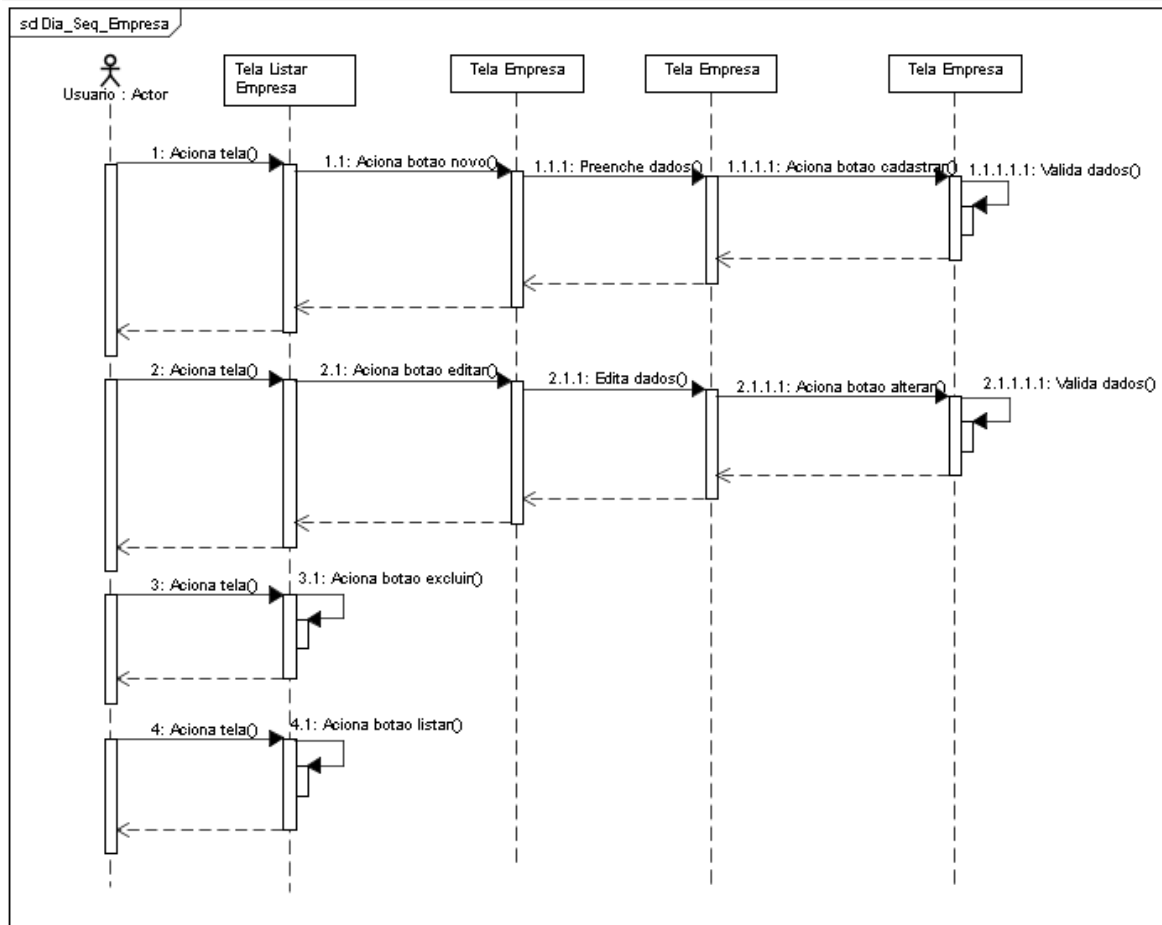


Diagrama de Sequência – Funcionário

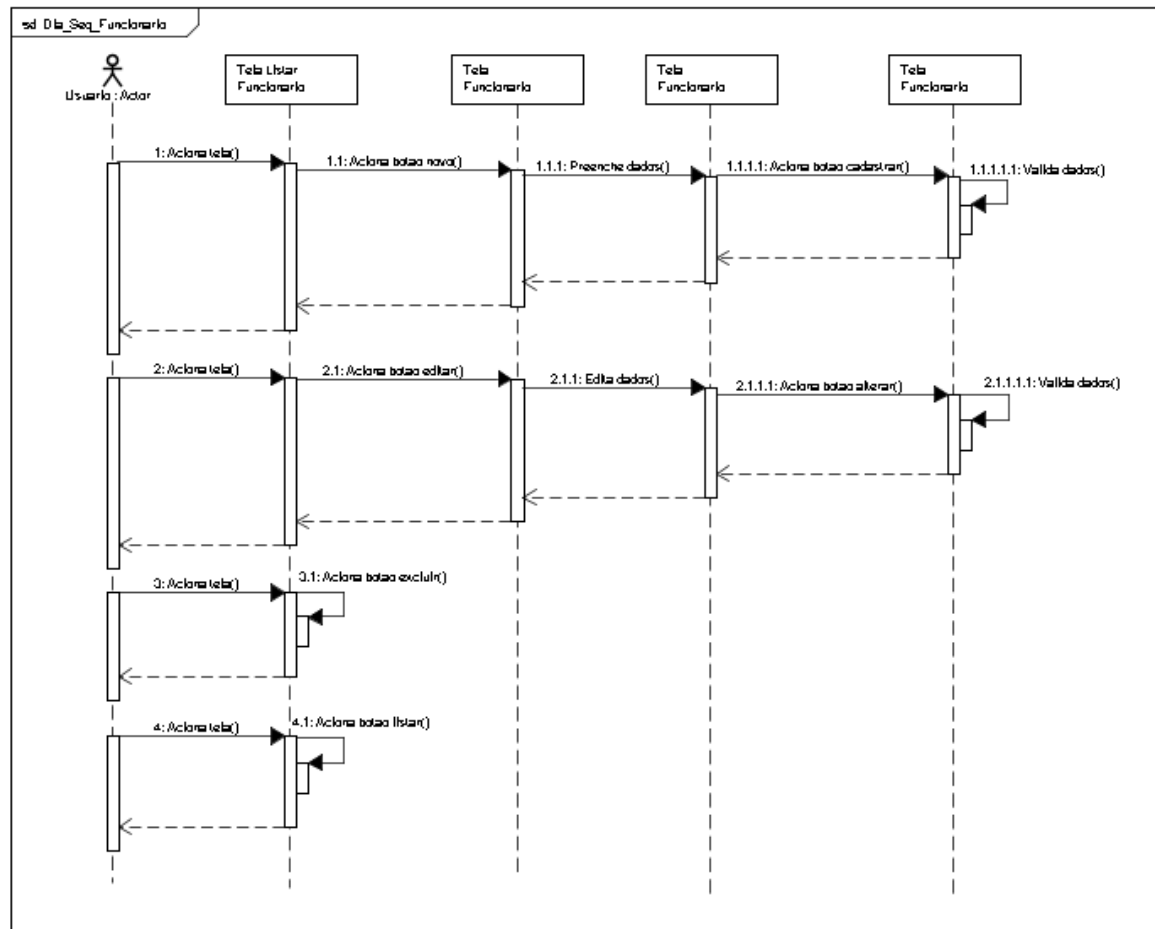


Diagrama de Sequência – Produto

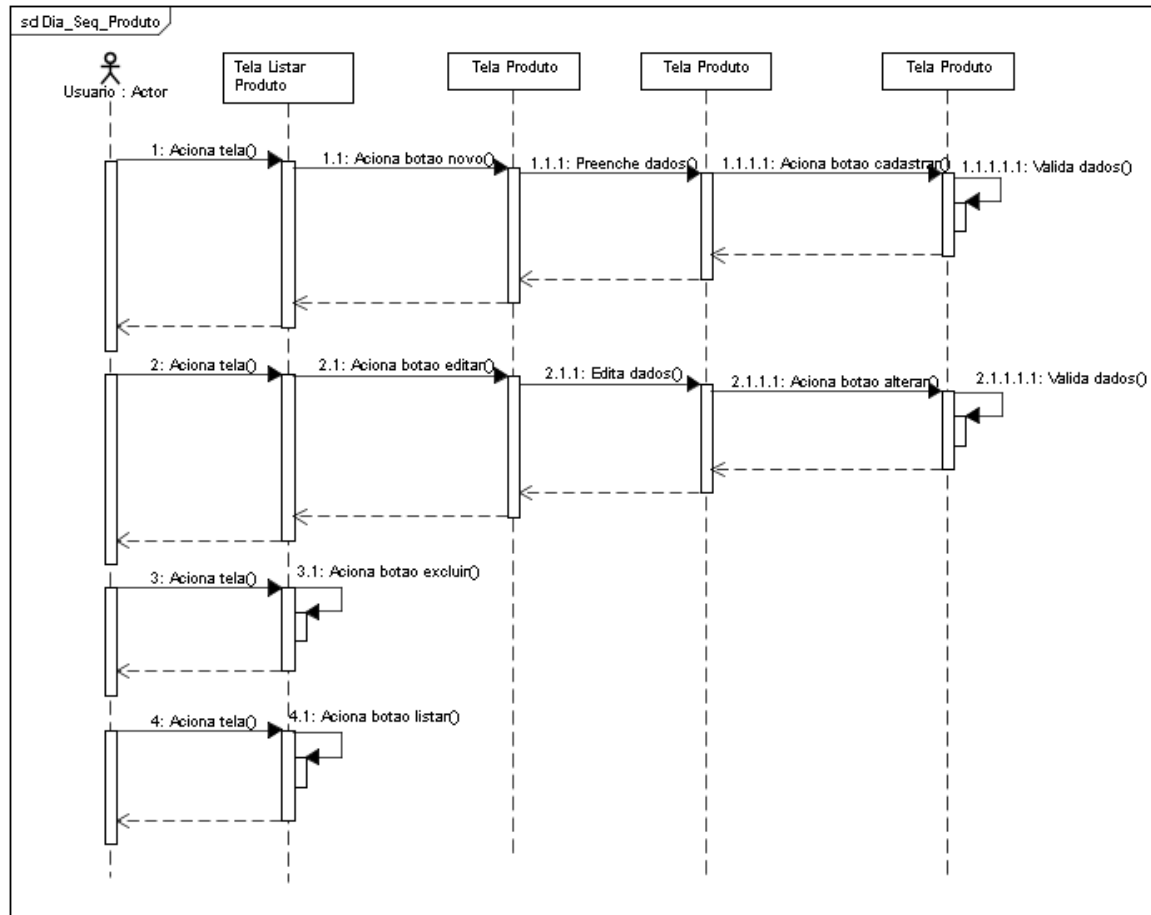


Diagrama de Sequência – Categoria

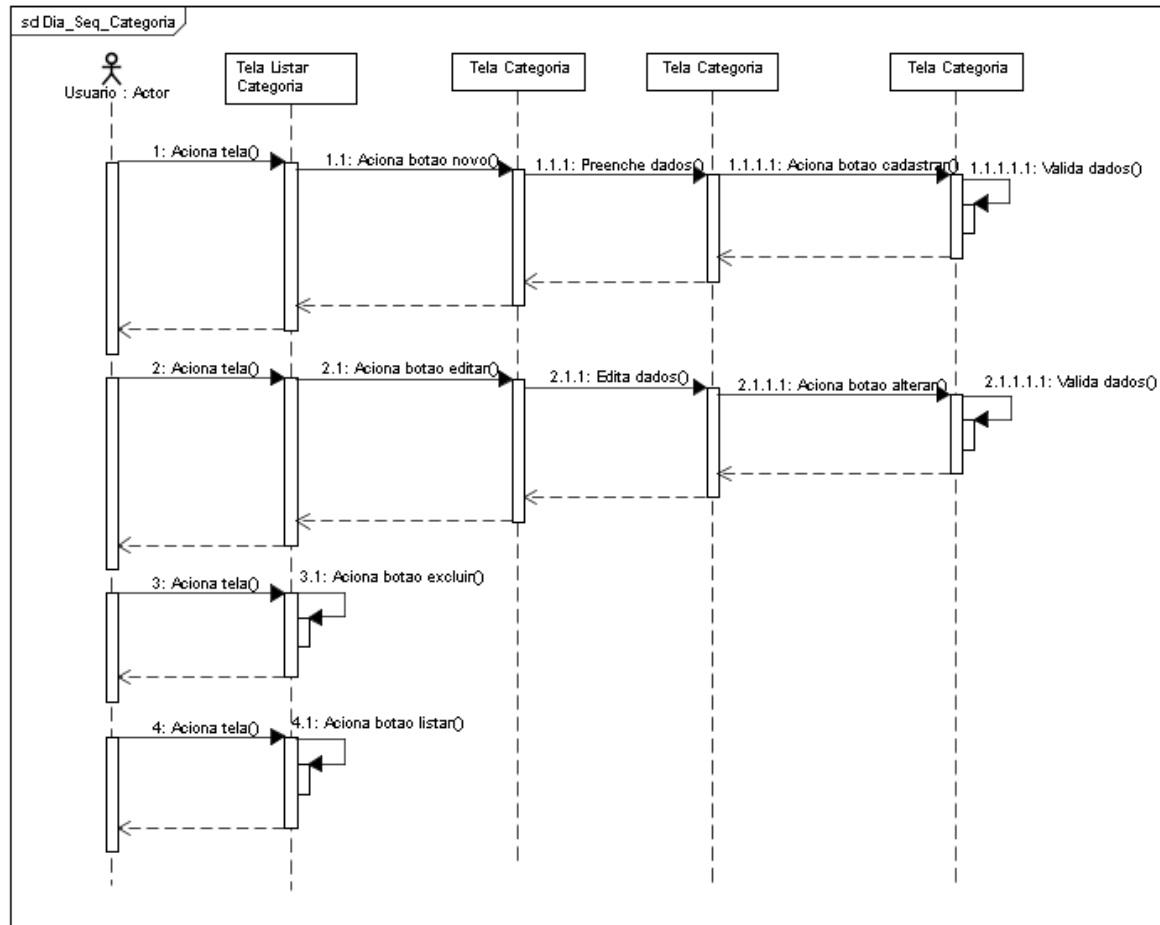


Diagrama de Sequência – Pais

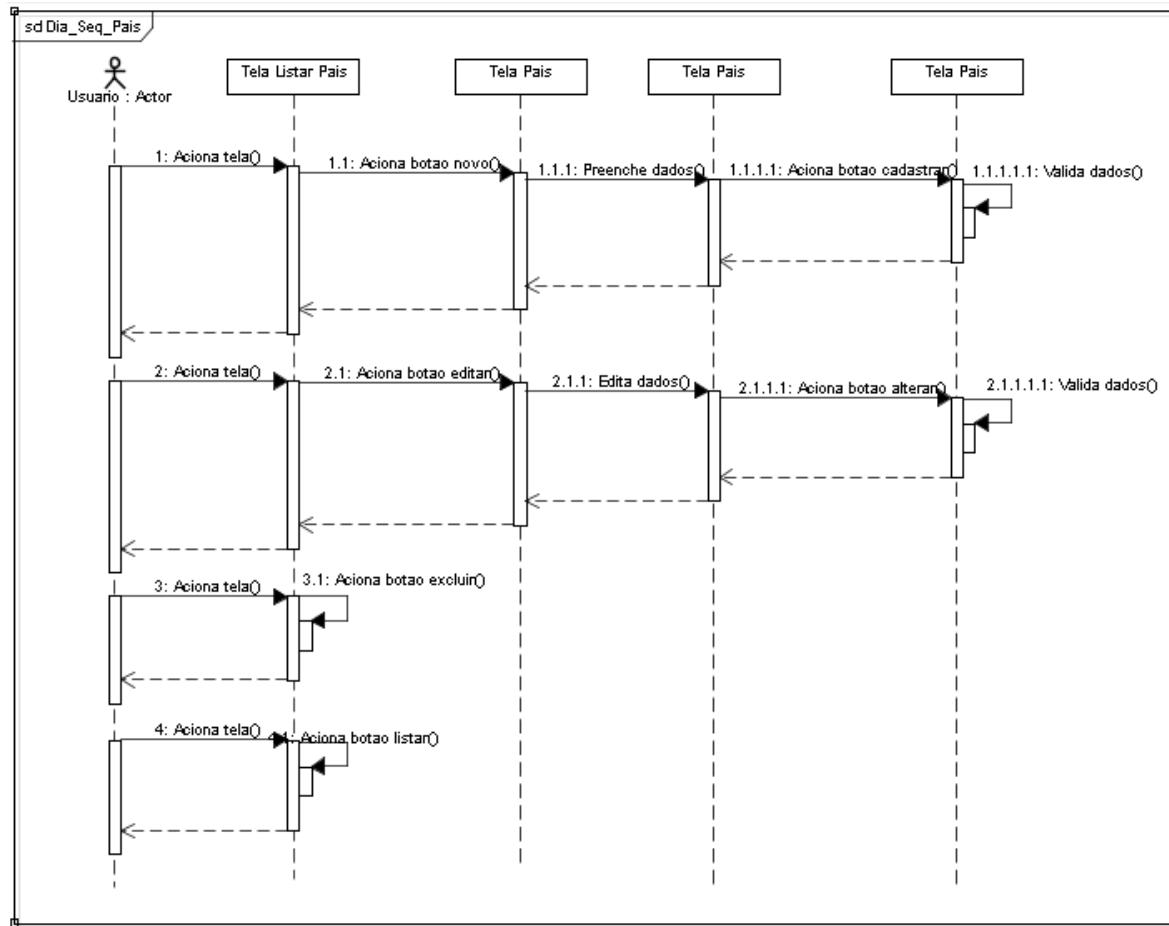


Diagrama de Sequência – Estado

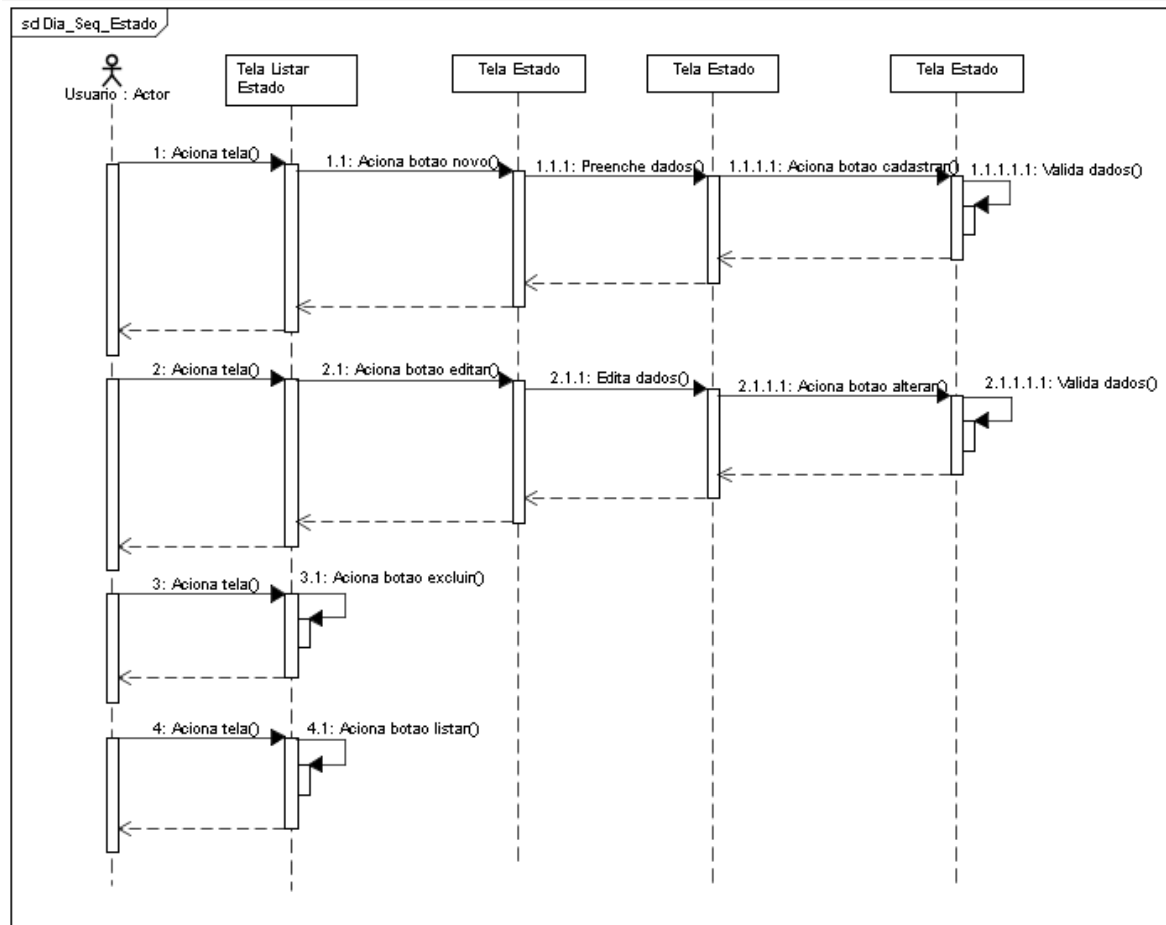


Diagrama de Sequência – Cidade

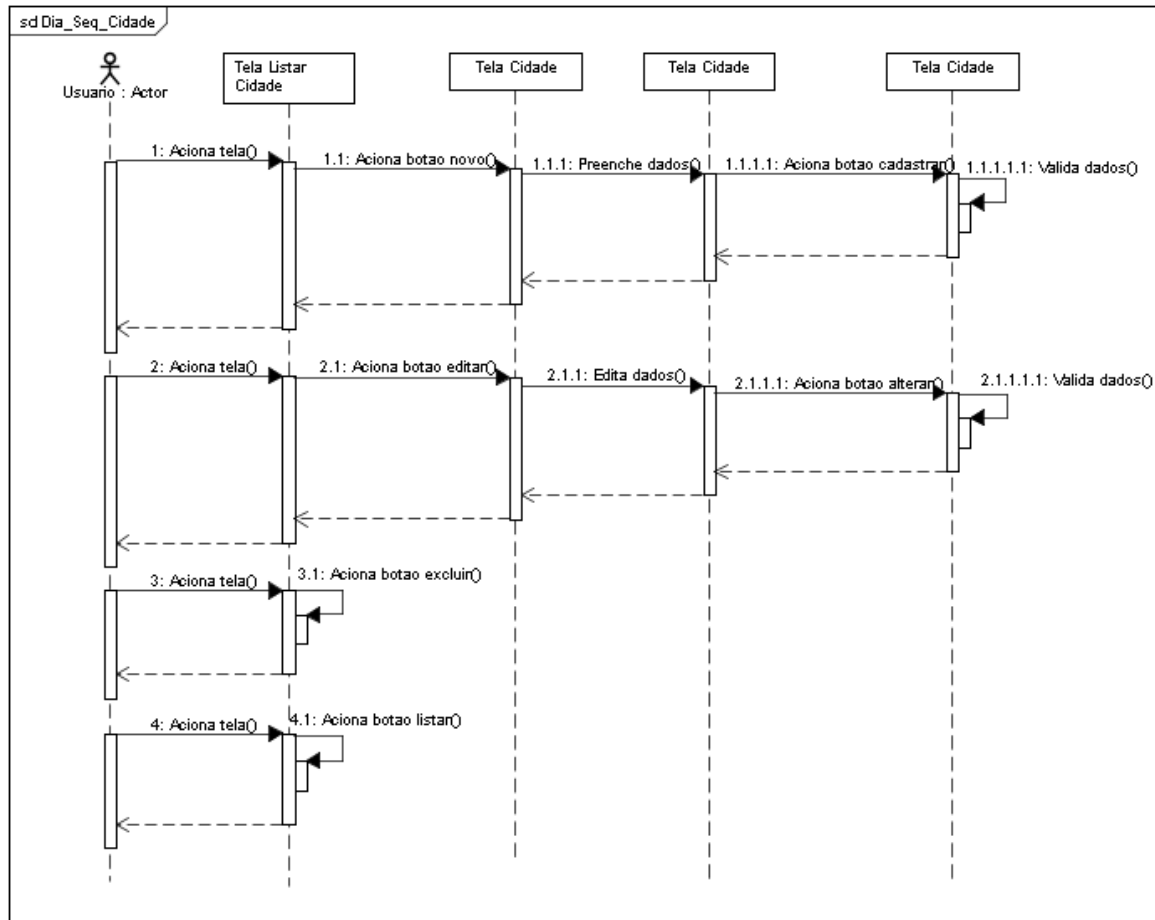


Diagrama de Sequência – Usuário

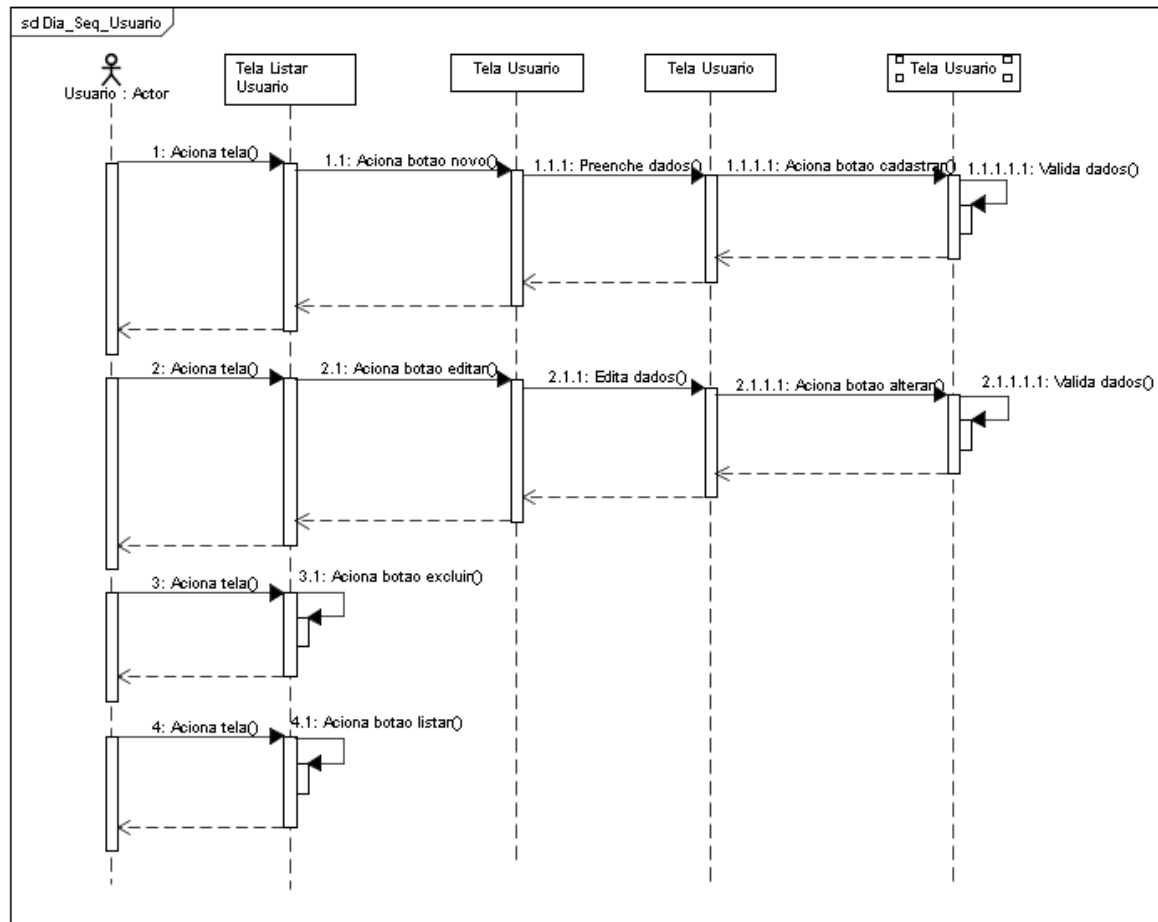


Diagrama de Sequência – Perfil

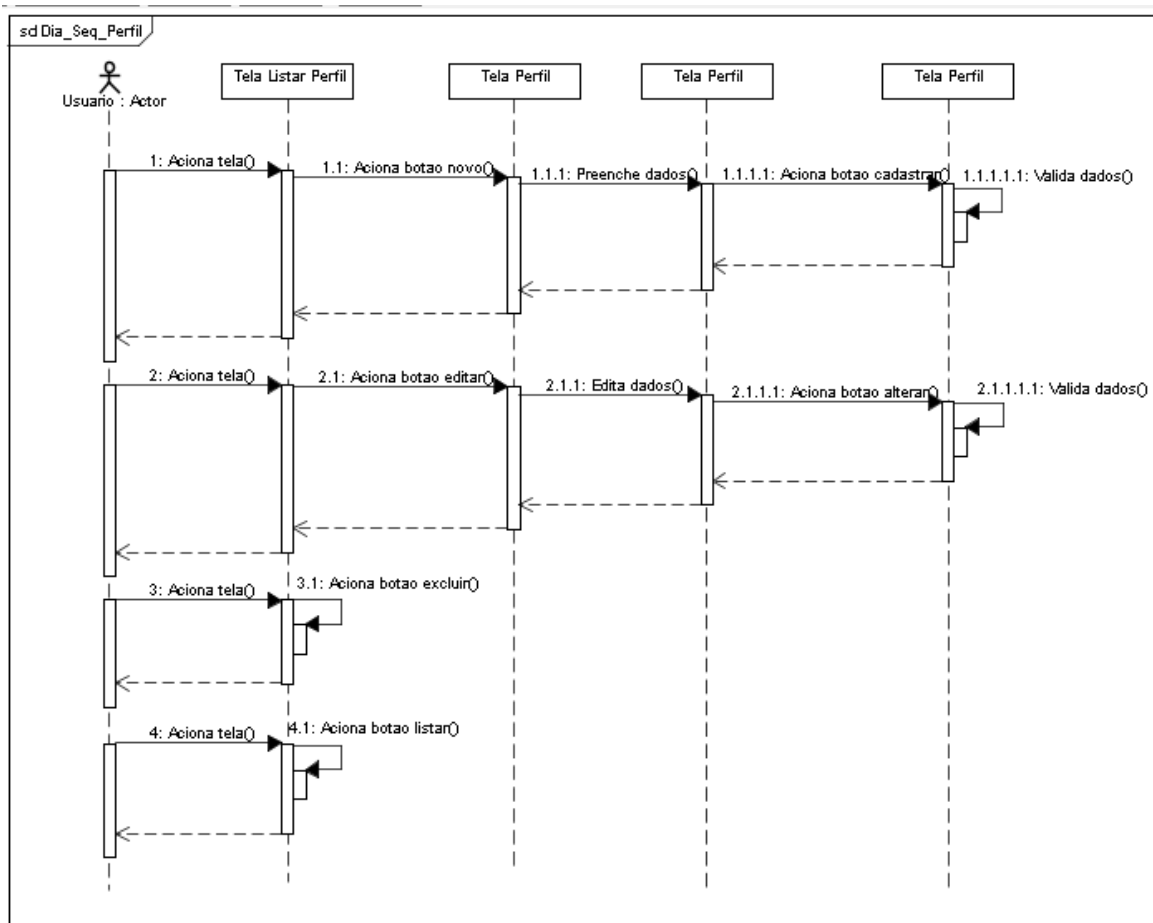


Diagrama de Sequência – Atividade

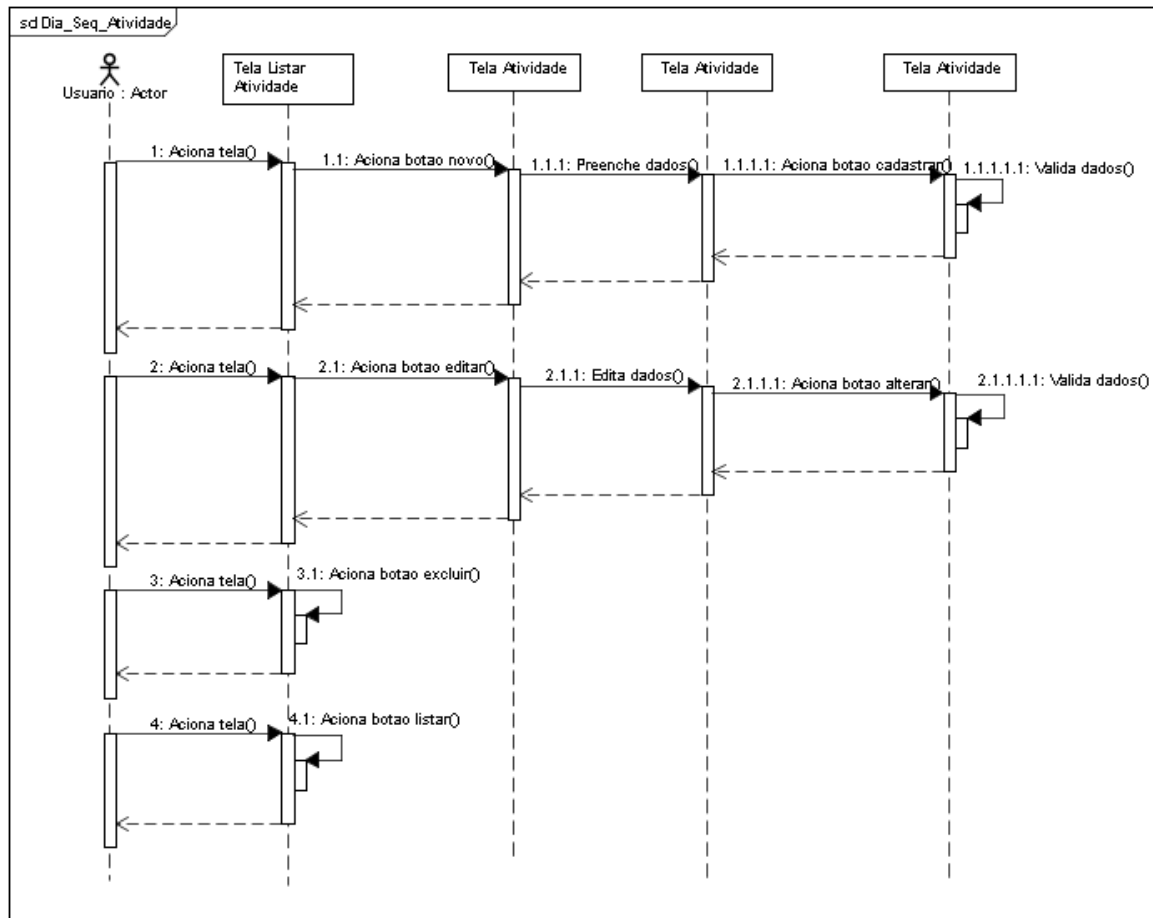


Diagrama de Sequência – Login

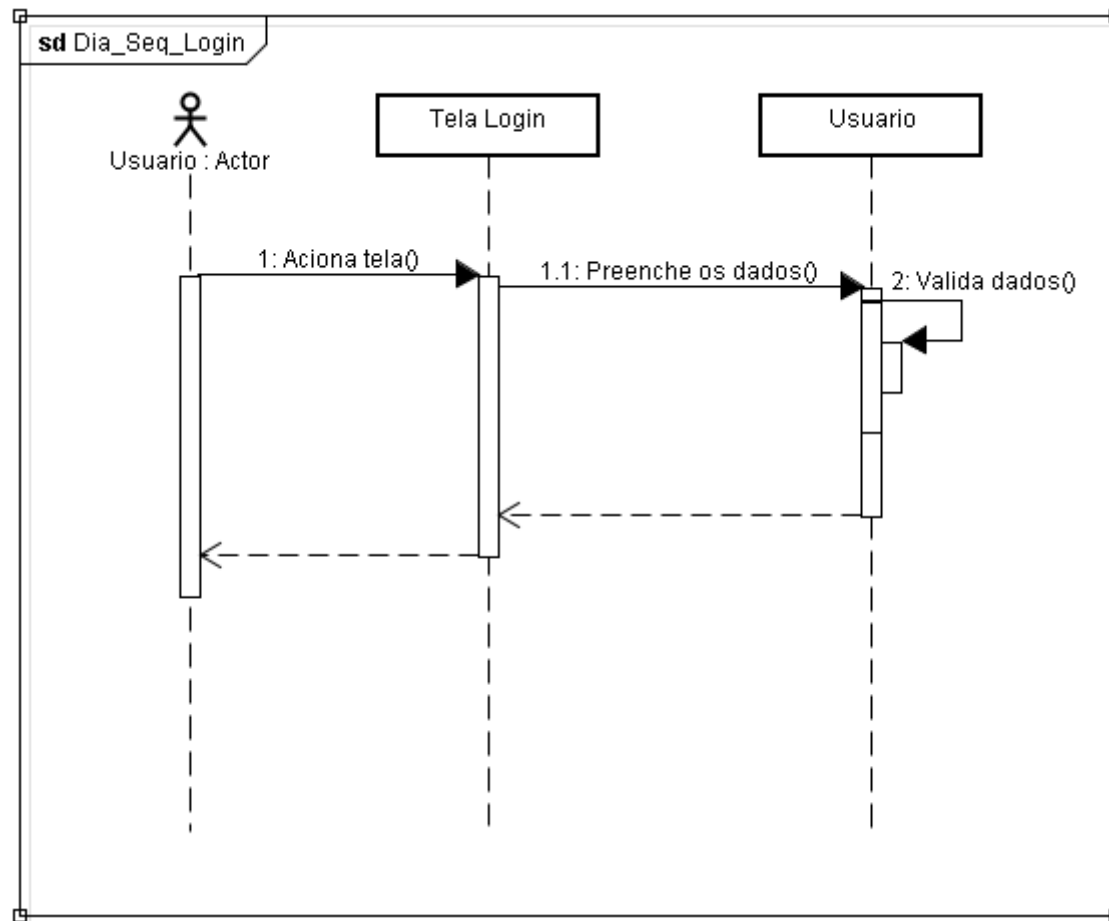
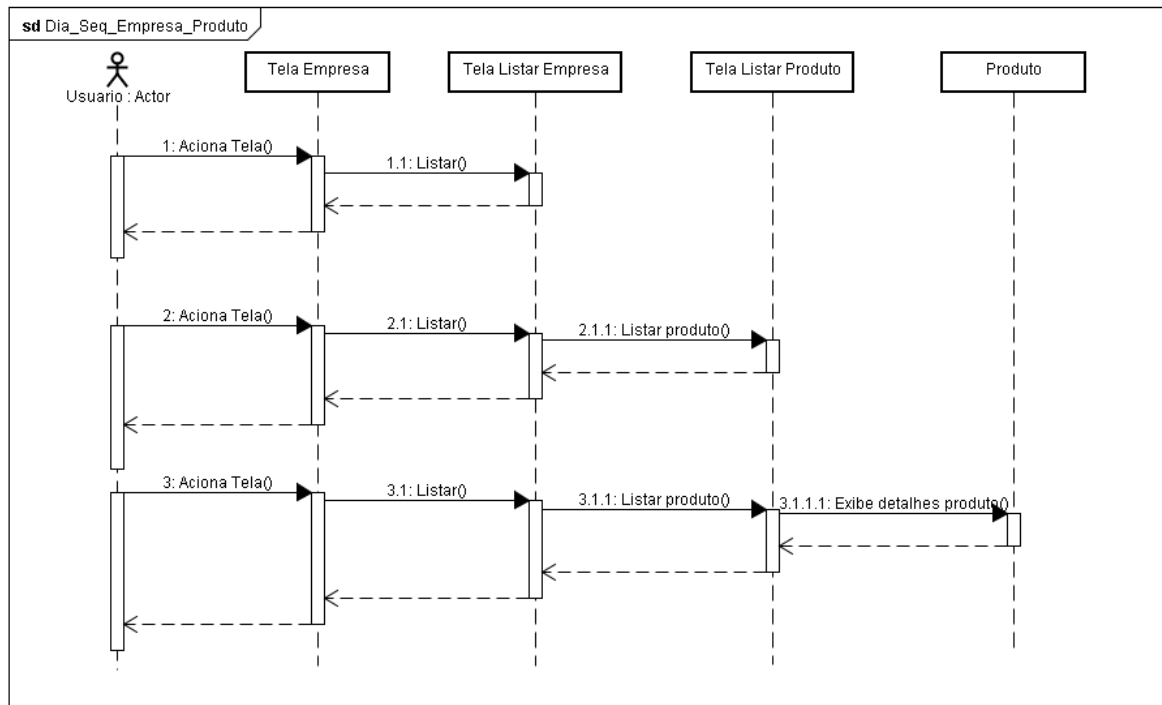


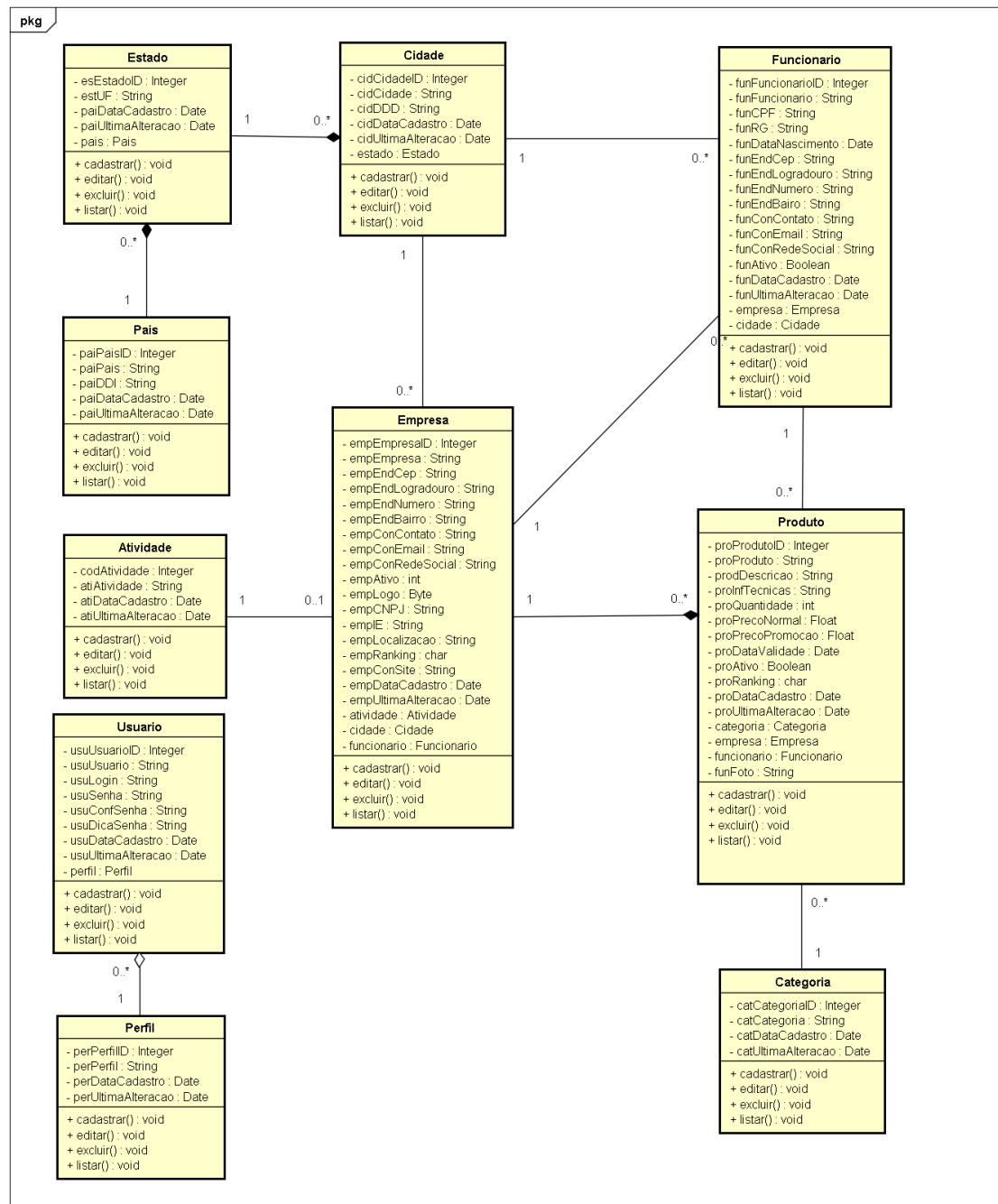
Diagrama de Sequência – Android Empresa, Produto e Detalhe do Produto



APÊNDICE L - 12 - Template Modelo de Objetos

Diagrama de Classes completo com atributos e métodos.

Versão 2.1



APÊNDICE M - 13 - Template Modelo Fisico de Dados

Esquema de banco de dados

Versão 2.1

```
CREATE SCHEMA `mbestpro_mbestproducts`;  
  
CREATE TABLE `mbestpro_mbestproducts`.`atividades` (  
  `atiAtividadeID` INT(10) NOT NULL AUTO_INCREMENT,  
  `atiAtividade` VARCHAR(50) NULL,  
  `atiDataCadastro` TIMESTAMP NULL DEFAULT NULL,  
  `atiUltimaAlteracao` TIMESTAMP NULL DEFAULT NULL,  
  PRIMARY KEY (`atiAtividadeID`)  
);  
  
CREATE TABLE `mbestpro_mbestproducts`.`paiss` (  
  `paiPaisID` INT(10) NOT NULL AUTO_INCREMENT ,  
  `paiPais` VARCHAR(50) NULL DEFAULT NULL,  
  `paiDDI` VARCHAR(10) NULL DEFAULT NULL,  
  `paiDataCadastro` TIMESTAMP NULL DEFAULT NULL ,  
  `paiUltimaAlteracao` TIMESTAMP NULL DEFAULT NULL,  
  PRIMARY KEY (`paiPaisID`));  
  
CREATE TABLE `mbestpro_mbestproducts`.`estados` (  
  `estEstadoID` INT(10) NOT NULL AUTO_INCREMENT,  
  `pais_PaiPaisID` INT(10) NOT NULL,  
  `estEstado` VARCHAR(50) NOT NULL,
```

```

`estUF` VARCHAR(10) NOT NULL,
`estDataCadastro` TIMESTAMP NULL DEFAULT NULL ,
`estUltimaAlteracao` TIMESTAMP NULL DEFAULT NULL ,
PRIMARY KEY (`estEstadoID`),
INDEX Estado_FKIndex1 (`pais_PaiPaisID`),
key fk_pais (`Pais_PaiPaisID`),
constraint fk_pais foreign key (`pais_PaiPaisID`)
references paiss (`paiPaisID`)
);

```

```

CREATE TABLE `mbestpro_mbestproducts`.`cidades` (
  `cidCidadeID` INT(10) NOT NULL AUTO_INCREMENT,
  `estados_EstEstadoID` INT(10) NOT NULL,
  `cidCidade` VARCHAR(50) NOT NULL,
  `cidDDD` VARCHAR(10) NOT NULL,
  `cidDataCadastro` TIMESTAMP NULL DEFAULT NULL ,
  `cidUltimaAlteracao` TIMESTAMP NULL DEFAULT NULL ,
  PRIMARY KEY (`cidCidadeID`),
  INDEX Cidades_FKIndex1 (`estados_EstEstadoID`),
  key fk_estado (`estados_EstEstadoID`),
  constraint fk_estado foreign key (`estados_EstEstadoID`)
  references estados (`estEstadoID`)
);

```

```

CREATE TABLE `mbestpro_mbestproducts`.`clientes` (
  `cliClienteID` INT(10) NOT NULL AUTO_INCREMENT ,
  `cliNome` VARCHAR(100) NOT NULL ,
  `cliCPF` VARCHAR(20) NOT NULL ,
  `cliDataNascimento` DATE NOT NULL ,
  `cliSexo` CHAR(1) NOT NULL ,

```

```

`cliEmail` VARCHAR(50) NOT NULL ,
`cliEndereco` VARCHAR(200) NULL DEFAULT NULL ,
`cliEndCEP` VARCHAR(15) NULL DEFAULT NULL ,
`cliEndNumero` INT(11) NULL DEFAULT NULL ,
`cliEndBairro` VARCHAR(100) NULL DEFAULT NULL ,
`cliEndTelefone` VARCHAR(20) NULL DEFAULT NULL ,
`cliEndCidade` VARCHAR(50) NULL DEFAULT NULL ,
`cliEndEstado` VARCHAR(50) NULL DEFAULT NULL ,
`cliEndPais` VARCHAR(50) NULL DEFAULT NULL ,
`cliEndComplemento` VARCHAR(250) NULL DEFAULT NULL ,
`cliDataCadastro` DATE NOT NULL ,
`cliUltimaAlteracao` DATE NOT NULL ,
PRIMARY KEY (`cliClienteID`)
);

```

```

CREATE TABLE `mbestpro_mbestproducts`.`perfils` (
`perPerfilID` INT(10) NOT NULL AUTO_INCREMENT ,
`perPerfil` VARCHAR(50) NULL DEFAULT NULL ,
`perDataCadastro` TIMESTAMP NULL DEFAULT NULL ,
`perUltimaAlteracao` TIMESTAMP NULL DEFAULT NULL ,
PRIMARY KEY (`perPerfilID`)
);

```

```

CREATE TABLE `mbestpro_mbestproducts`.`usuarios` (
`usuUsuarioID` INT(10) NOT NULL AUTO_INCREMENT ,
`usuLogin` VARCHAR(50) NOT NULL ,
`usuSenha` VARCHAR(40) NOT NULL ,
`usuMatricula` INT(11) NOT NULL ,
`usuNome` VARCHAR(100) NOT NULL ,
`usuEmail` VARCHAR(100) NOT NULL ,
`usuDicaSenha` VARCHAR(50) NULL DEFAULT NULL ,

```

```

`usuConfSenha` VARCHAR(50) NULL DEFAULT NULL ,
`usuAtivo` TINYINT(1) NULL DEFAULT NULL ,
`usuTentativaLogin` INT(11) NULL DEFAULT NULL ,
`usuDataCadastro` TIMESTAMP NULL DEFAULT NULL ,
`usuUltimaAlteracao` TIMESTAMP NULL DEFAULT NULL ,
`perfil_PerPerfillID` INT(10) NOT NULL ,
PRIMARY KEY (`usuUsuariID`),
INDEX `Usuarios_FKIndex1` (`perfil_PerPerfillID` ASC) ,
INDEX `fk_perfis` (`perfil_PerPerfillID` ASC) ,
CONSTRAINT `fk_perfis`
  FOREIGN KEY (`perfil_PerPerfillID` )
  REFERENCES `mbestpro_mbestproducts`.`perfis` (`perPerfillID` )
);

```

```

CREATE TABLE `mbestpro_mbestproducts`.`categorias` (
  `catCategorialID` INT(10) NOT NULL AUTO_INCREMENT,
  `catCategoria` VARCHAR(50) NULL DEFAULT NULL,
  `catDataCadastro` TIMESTAMP NULL DEFAULT NULL,
  `catUltimaAlteracao` TIMESTAMP NULL DEFAULT NULL,
  PRIMARY KEY (`catCategorialID`)
);

```

```

CREATE TABLE `mbestpro_mbestproducts`.`empresas` (
  `empEmpresaID` INT(10) NOT NULL AUTO_INCREMENT,
  `atividades_AtividadeID` INT(10) NOT NULL,
  `cidades_CidCidadeID` INT(10) NOT NULL,
  `empEmpresa` VARCHAR(50) NULL,
  `empCNPJ` VARCHAR(25) NULL,
  `empIE` VARCHAR(25) NULL,
  `empEndCEP` VARCHAR(25) NULL,

```

```

`empEndLogradouro` VARCHAR(50) NULL,
`empEndBairro` VARCHAR(25) NULL,
`empEndNumero` VARCHAR(25) NULL,
`empEndComplemento` VARCHAR(250) NULL,
`empConContato` VARCHAR(25) NULL,
`empConEmail` VARCHAR(50) NULL,
`empConRedeSocial` VARCHAR(250) NULL,
`empConSite` VARCHAR(50) NULL,
`empLocalizacao` VARCHAR(100) NULL,
`empLogo` VARCHAR(50) NULL DEFAULT NULL,
`empDataCadastro` TIMESTAMP NULL DEFAULT NULL,
`empUltimaAlteracao` TIMESTAMP NULL DEFAULT NULL,
`empAtivo` TINYINT(1) NULL DEFAULT NULL ,
`empRanking` CHAR(10) NULL,
PRIMARY KEY(`empEmpresaID`),
INDEX Empresas_FKIndex2(`cidades_CidCidadeID`),
INDEX Empresas_FKIndex3(`atividades_AtiAtividadeID`),
key fk_cidade (`cidades_CidCidadeID`),
constraint fk_cidade foreign key (`cidades_CidCidadeID`)
references cidades (`cidCidadeID`),

key fk_atividades (`atividades_AtiAtividadeID`),
constraint fk_atividades foreign key (`atividades_AtiAtividadeID`)
references atividades (`atiAtividadeID`)
);

```

```

CREATE TABLE `mbestpro_mbestproducts`.`funcionarios` (
`funFuncionarioID` INT(10) NOT NULL AUTO_INCREMENT,
`cidades_CidCidadeID` INT(10) NOT NULL,
`empresas_EmpEmpresaID` INT(10) NOT NULL,
`funFuncionario` VARCHAR(50) NULL DEFAULT NULL,

```

```

`funCPF` VARCHAR(25) NULL DEFAULT NULL,
`funRG` VARCHAR(25) NULL DEFAULT NULL,
`funDataNascimento` TIMESTAMP NULL DEFAULT NULL,
`funEndCEP` VARCHAR(25) NULL DEFAULT NULL,
`funEndLogradouro` VARCHAR(50) NULL DEFAULT NULL,
`funEndBairro` VARCHAR(25) NULL DEFAULT NULL,
`funEndNumero` VARCHAR(25) NULL DEFAULT NULL,
`funEndComplemento` VARCHAR(250) NULL DEFAULT NULL,
`funConContato` VARCHAR(25) NULL DEFAULT NULL,
`funConEmail` VARCHAR(50) NULL DEFAULT NULL,
`funConRedeSocial` VARCHAR(100) NULL DEFAULT NULL,
`funAtivo` TINYINT(1) NULL DEFAULT NULL ,
`funDataCadastro` TIMESTAMP NULL DEFAULT NULL,
`funUltimaAlteracao` TIMESTAMP NULL DEFAULT NULL,
PRIMARY KEY(`FunFuncionarioID`),
INDEX Funcionarios_FKIndex2(`empresas_EmpEmpresalD`),
INDEX Funcionarios_FKIndex3(`cidades_CidCidadeID`),
  key fk_empresa (`empresas_EmpEmpresalD`),
  constraint fk_empresa foreign key (`empresas_EmpEmpresalD`)
    references empresas (`empEmpresalD`),

  key fk_cidadeFun (`cidades_CidCidadeID`),
  constraint fk_cidadeFun foreign key (`cidades_CidCidadeID`)
    references cidades (`cidCidadeID`)

);

```

```

CREATE TABLE `mbestpro_mbestproducts`.`produtos` (
  `proProdutoID` INT(10) NOT NULL AUTO_INCREMENT,
  `funcionarios_FunFuncionarioID` INT(10) NOT NULL,

```



```

`empresas_EmpEmpresaID` INT(10) NOT NULL,
`categorias_CatCategorialID` INT(10) NOT NULL,
`proProduto` VARCHAR(50) NULL DEFAULT NULL,
`proDescricao` VARCHAR(100) NULL DEFAULT NULL,
`proInfTecnicas` VARCHAR(1000) NULL DEFAULT NULL,
`proQuantidade` INT(10) NULL DEFAULT NULL,
`proPrecoNormal` FLOAT NULL DEFAULT NULL,
`proPrecoPromocao` FLOAT NULL DEFAULT NULL,
`proDataValidade` TIMESTAMP NULL DEFAULT NULL,
`proRanking` CHAR(10) NULL DEFAULT NULL,
`proAtivo` TINYINT(1) NULL DEFAULT NULL,
`proDataCadastro` TIMESTAMP NULL DEFAULT NULL,
`proUltimaAlteracao` TIMESTAMP NULL DEFAULT NULL,
`proFoto` VARCHAR(50) NULL DEFAULT NULL,
PRIMARY KEY(`ProProdutoID`),
INDEX Produto_FKIndex1(`categorias_CatCategorialID`),
INDEX Produto_FKIndex2(`empresas_EmpEmpresaID`),
INDEX Produtos_FKIndex4(`funcionarios_FunFuncionariID`),
  key fk_categoria (`categorias_CatCategorialID`),
  constraint fk_categoria foreign key (`categorias_CatCategorialID`)
    references categorias (`catCategorialID`),

  key fk_empresaPro (`empresas_EmpEmpresaID`),
  constraint fk_empresaPro foreign key (`empresas_EmpEmpresaID`)
    references empresas (`empEmpresaID`),

  key fk_funcionarioPro (`funcionarios_FunFuncionariID`),
  constraint fk_funcionarioPro foreign key (`funcionarios_FunFuncionariID`)
    references funcionarios (`FunFuncionariID`)
);

```

```

INSERT INTO mbestpro_mbestproducts.perfis(perPerfil,perDataCadastro,perUltimaAlteracao) VALUES( 'ROLE_ADMIN','2013-01-01','2013-01-01');
INSERT INTO mbestpro_mbestproducts.perfis(perPerfil,perDataCadastro,perUltimaAlteracao) VALUES( 'ROLE_GERENTE','2013-01-01','2013-01-01');
INSERT INTO mbestpro_mbestproducts.perfis(perPerfil,perDataCadastro,perUltimaAlteracao) VALUES( 'ROLE_TECNICO','2013-01-01','2013-01-01');

```

-- login

```

INSERT INTO mbestpro_mbestproducts.usuarios
(usuLogin,usuSenha,usuMatricula,usuNome,usuEmail,usuTentativaLogin,perfil_PerPerfilID)
VALUES('admin','202cb962ac59075b964b07152d234b70',368923,'Maicon Fang','admin@mbestpro_mbestproducts.com.br',0 ,1);

```

```

INSERT INTO mbestpro_mbestproducts.usuarios
(usuLogin,usuSenha,usuMatricula,usuNome,usuEmail,usuTentativaLogin,perfil_PerPerfilID)
VALUES('joaquim','827ccb0eea8a706c4c34a16891f84e7b',2432109,'joaquim silva','joaquim@mbestpro_mbestproducts.com.br',1
,2);

```

```

INSERT INTO mbestpro_mbestproducts.usuarios
(usuLogin,usuSenha,usuMatricula,usuNome,usuEmail,usuTentativaLogin,perfil_PerPerfilID)
VALUES('roberto','202cb962ac59075b964b07152d234b70',21374323,'roberto
simão','roberto@mbestpro_mbestproducts.com.br',1,3);

```

-- cliente

```

INSERT INTO mbestpro_mbestproducts.clientes
(cliNome,cliCPF,cliDataNascimento,cliSexo,cliEmail,cliDataCadastro,cliUltimaAlteracao)
VALUES ('JON BON JOVI','956.845.65-68', '1965-02-01','2','jon@jovi.com','2013-06-10','2013-06-10');

```

```

INSERT INTO mbestpro_mbestproducts.clientes
(cliNome,cliCPF,cliDataNascimento,cliSexo,cliEmail,cliDataCadastro,cliUltimaAlteracao)
VALUES ('DAVI','888.845.85-55','1985-10-10','2','davi@davi.com','2013-06-11','2013-06-11');

```

```
INSERT INTO mbestpro_mbestproducts.clientes
(cliNome,cliCPF,cliDataNascimento,cliSexo,cliEmail,cliDataCadastro,cliUltimaAlteracao)
VALUES ('Maicon','737.764.422-20', '1985-10-10','2','maicon@gmail.com','2013-06-11','2013-06-11');
```

```
-- pais
```

```
INSERT INTO mbestpro_mbestproducts.paiss(paiPais, paiDDI, paiDataCadastro, paiUltimaAlteracao)
VALUES('Brazil','55','2013-01-01','2013-01-01');
```

```
INSERT INTO mbestpro_mbestproducts.paiss(paiPais, paiDDI, paiDataCadastro, paiUltimaAlteracao)
VALUES('Austrália','78','2013-01-01','2013-01-01');
```

```
INSERT INTO mbestpro_mbestproducts.paiss(paiPais, paiDDI, paiDataCadastro, paiUltimaAlteracao)
VALUES('Canada','11','2013-01-01','2013-01-01');
```

```
-- estado
```

```
INSERT INTO mbestpro_mbestproducts.estados(estEstado, pais_PaiPaisID ,estUF, estDataCadastro, estUltimaAlteracao)
VALUES('Paraná',1, 'PR','2013-01-01','2013-01-01');
```

```
INSERT INTO mbestpro_mbestproducts.estados(estEstado, pais_PaiPaisID ,estUF, estDataCadastro, estUltimaAlteracao)
VALUES('Santa Catarina',1, 'CD','2013-01-01','2013-01-01');
```

```
INSERT INTO mbestpro_mbestproducts.estados(estEstado, pais_PaiPaisID ,estUF, estDataCadastro, estUltimaAlteracao)
VALUES('Bahia',1, 'CD','2013-01-01','2013-01-01');
```

```
-- cidade
```

```
INSERT INTO mbestpro_mbestproducts.cidades(cidCidade, estados_EstEstadoID ,cidDDD, cidDataCadastro, cidUltimaAlteracao)
VALUES('Curitiba',1, '41','2013-01-01','2013-01-01');
```

```
INSERT INTO mbestpro_mbestproducts.cidades(cidCidade, estados_EstEstadoID ,cidDDD, cidDataCadastro, cidUltimaAlteracao)
VALUES('Santa Helena',1, '45','2013-01-01','2013-01-01');
```

```
INSERT INTO mbestpro_mbestproducts.cidades(cidCidade, estados_EstEstadoID ,cidDDD, cidDataCadastro, cidUltimaAlteracao)
VALUES('Londrina',1, '45','2013-01-01','2013-01-01');
```

```
-- atividade
```

```
INSERT INTO mbestpro_mbestproducts.atividades(atiAtividade, atiDataCadastro, atiUltimaAlteracao)
VALUES('Vende Eletronicos','2013-01-01','2013-01-01');
```

```
INSERT INTO mbestpro_mbestproducts.atividades(atiAtividade, atiDataCadastro, atiUltimaAlteracao)
VALUES('Vende Roupas','2013-01-01','2013-01-01');
```

```
INSERT INTO mbestpro_mbestproducts.atividades(atiAtividade, atiDataCadastro, atiUltimaAlteracao)
VALUES('Construção Civil','2013-01-01','2013-01-01');
```

```
-- categoria
```

```
INSERT INTO mbestpro_mbestproducts.categorias(catCategoria, catDataCadastro, catUltimaAlteracao)
VALUES('Vestuário e acessórios','2013-01-01','2013-01-01');
```

```
INSERT INTO mbestpro_mbestproducts.categorias(catCategoria, catDataCadastro, catUltimaAlteracao)
VALUES('Mídia','2013-01-01','2013-01-01');
```

```
INSERT INTO mbestpro_mbestproducts.categorias(catCategoria, catDataCadastro, catUltimaAlteracao)
VALUES('Construção Civil','2013-01-01','2013-01-01');
```

```
-- empresas
```

```
INSERT INTO mbestpro_mbestproducts.empresas(empEmpresa, atividades_AtiAtividadeID ,cidades_CidCidadeID, empCNPJ,
empIE, empEndCEP,
empEndLogradouro, empEndBairro, empEndNumero,
empEndComplemento, empConContato, empConEmail, empConRedeSocial,
```

```
empDataCadastro, empUltimaAlteracao, empAtivo, empRanking)
VALUES('Casol', 3 ,1, '21528607000119', '5196893247', '80230000',
'4199447855', 'pedro.silva@casol.com.br', 'pedro silva',
01-01','2013-01-01', 1, '5');
```

```
INSERT INTO mbestpro_mbestproducts.empresas(empEmpresa, atividades_AtividadeID ,cidades_CidCidadeID, empCNPJ,
empIE, empEndCEP,
empEndLogradouro, empEndBairro, empEndNumero,
empEndComplemento, empConContato, empConEmail, empConRedeSocial,
empConSite, empLocalizacao, empLogo,
empDataCadastro, empUltimaAlteracao, empAtivo, empRanking)
VALUES('Renner ', 2 ,3, '09045606000118', '6155311116', '86081-186',
'UEL', '43', 'renner@renner.com.br', 'renner',
'2013-01-01','2013-01-01', 1, 3);
```

```
INSERT INTO mbestpro_mbestproducts.empresas(empEmpresa, atividades_AtividadeID ,cidades_CidCidadeID, empCNPJ,
empIE, empEndCEP,
empEndLogradouro, empEndBairro, empEndNumero,
empEndComplemento, empConContato, empConEmail, empConRedeSocial,
empConSite, empLocalizacao, empLogo,
empDataCadastro, empUltimaAlteracao, empAtivo, empRanking)
VALUES('Havan ', 2 ,3, '04164328000102', '3302347840', '80020904',
'havan@havan.com.br', 'havan',
01-01','2013-01-01', 1, 8);
```

```
empConSite, empLocalizacao, empLogo,
'Avenida Barsil', 'Centro', '1580', 'Proximo a rodoviaria',
'www.casol.com.br', 'boa', 'caminho da imagem', '2013-
empEndLogradouro, empEndBairro, empEndNumero,
empConSite, empLocalizacao, empLogo,
'Rua Aparecido da Silva', 'Jardim Continental', '819',
'www.renner.com.br', 'ótima', 'caminho da imagem',
empEndLogradouro, empEndBairro, empEndNumero,
empConSite, empLocalizacao, empLogo,
'Avenida Luiz Xavier 66', 'Centro', '492', 'UEL', '43',
'www.havan.com.br', 'ruim', 'caminho da imagem', '2013-
```

-- funcionario

```
INSERT INTO mbestpro_mbestproducts.funcionarios(funFuncionario, cidades_CidCidadeID, empresas_EmpEmpresaID, funCPF,
funRG, funDataNascimento,
funEndNumero, funEndComplemento, funConContato, funConEmail,
funUltimaAlteracao)
VALUES('Nick', 1, 2, '27816055579', '16.146.389-7', '1984-01-01',
'859', 'Escola municipal', '41994528', 'nick@renner.com.br',
'68627270', 'Rua São Bento', 'Jardim Atlântico',
'nick RS', 1, '2013-01-01','2013-01-01');
```

```
INSERT INTO mbestpro_mbestproducts.funcionarios(funFuncionario, cidades_CidCidadeID, empresas_EmpEmpresaID, funCPF,
funRG, funDataNascimento,
funEndNumero, funEndComplemento, funConContato, funConEmail,
funUltimaAlteracao)
VALUES('Henrique Otávio Juan Barros', 1, 2, '73796183778', '199840891', '1980-02-07',
'85320000', 'Av Silva Jardim', 'Centro', '458', 'posto
de combustivel', '4199554722', 'henrique.otavio@renner.com.br',
'nao tem', 1, '2013-01-01','2013-01-01');
```

```
INSERT INTO mbestpro_mbestproducts.funcionarios(funFuncionario, cidades_CidCidadeID, empresas_EmpEmpresaID, funCPF,
funRG, funDataNascimento,
funEndNumero, funEndComplemento, funConContato, funConEmail,
funUltimaAlteracao)
VALUES('Pedro Antonio', 1, 3, '24766135504', '470032431', '1974-02-07',
```

```
'Farmacia Vita', '4199664218', 'pedro.antonio@havan.com.br',
'8582000', 'Av Silva Jardim', 'Rebouças', '1987',
'nao tem', 1, '2013-01-01','2013-01-01');
```

```
-- produtos
```

```
INSERT INTO mbestpro_mbestproducts.produtos(proProduto, funcionarios_FunFuncionarioID, empresas_EmpEmpresalD,
categorias_CatCategorialD,
proDescricao, proInfTecnicas, proQuantidade,
proPrecoNormal, proPrecoPromocao,
proDataValidade, proRanking, proAtivo, proDataCadastro,
proUltimaAlteracao, proFoto)
VALUES('Camisa Polo', 1, 1, 1,
'Camisa polo cor verde com faixas amarelas', 'Origem
brasileira, boa malha', 20, 200.00, 100.00,
'2016-01-01', '1', 1, '2013-01-01','2013-01-01', 'Foto da
camisa polo');
```

```
INSERT INTO mbestpro_mbestproducts.produtos(proProduto, funcionarios_FunFuncionarioID, empresas_EmpEmpresalD,
categorias_CatCategorialD,
proDescricao, proInfTecnicas, proQuantidade,
proPrecoNormal, proPrecoPromocao,
proDataValidade, proRanking, proAtivo, proDataCadastro,
proUltimaAlteracao, proFoto)
VALUES('Jaqueta Osmose', 1, 1, 1,
'Jaqueta osmoze para intenso inverno', 'Origem canadense,
ótima malha', 10, 600.00, 500.00,
'2016-01-01', '1', 1, '2013-01-01','2013-01-01', 'Foto da
Jaqueta Osmoze');
```

```

INSERT INTO mbestpro_mbestproducts.produtos(proProduto, funcionarios_FunFuncionarioID, empresas_EmpEmpresaID,
categorias_CatCategorialID,
proDescricao, proInfTecnicas, proQuantidade,
proPrecoNormal, proPrecoPromocao,
proDataValidade, proRanking, proAtivo, proDataCadastro,
proUltimaAlteracao, proFoto)
VALUES('Meia Adidas', 1, 1, 1,
'Meia adidas para intenso inverno', 'Origem EUA, ótima
malha', 100, 25.00, 15.00,
'2013-01-01', '1', 1, '2013-01-01','2013-01-01', 'Foto da
meia');

```

APÊNDICE N - 14 - Template Diagramas Suplementares

Diagramas de Transição de Estados

Versão 2.1

Histórico da Revisão

Data	Versão	Descrição	Autor
10/08/2013	1.0	Versão inicial	Maicon Alexander Illes Fang
11/12/2013	2.0	Versão secundaria	Maicon Alexander Illes Fang
25/02/2014	2.1	Ajustes e inserção de novos diagramas de	Maicon Alexander Illes

Diagrama de Transições de Estado – Empresa

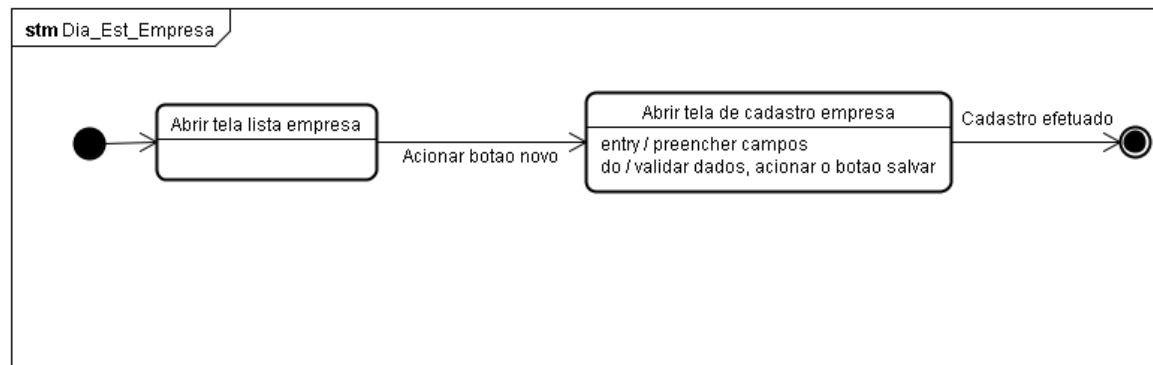


Diagrama de Transições de Estado – Funcionário

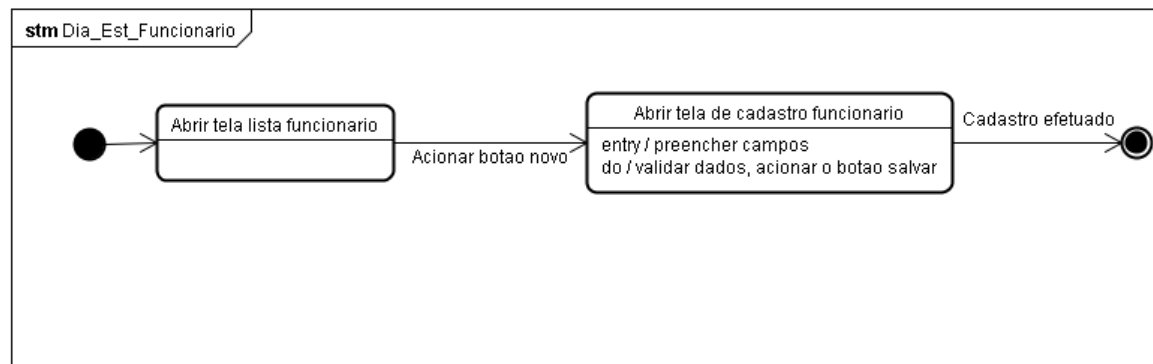


Diagrama de Transições de Estado – Produto

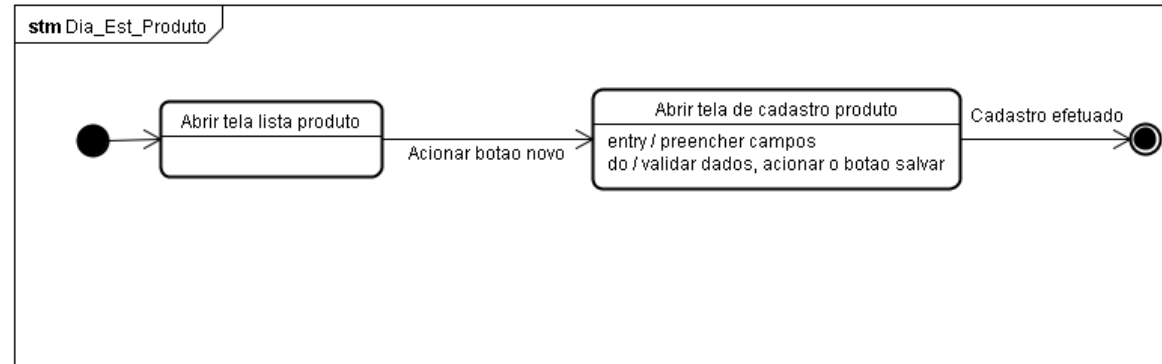


Diagrama de Transições de Estado – Categoria

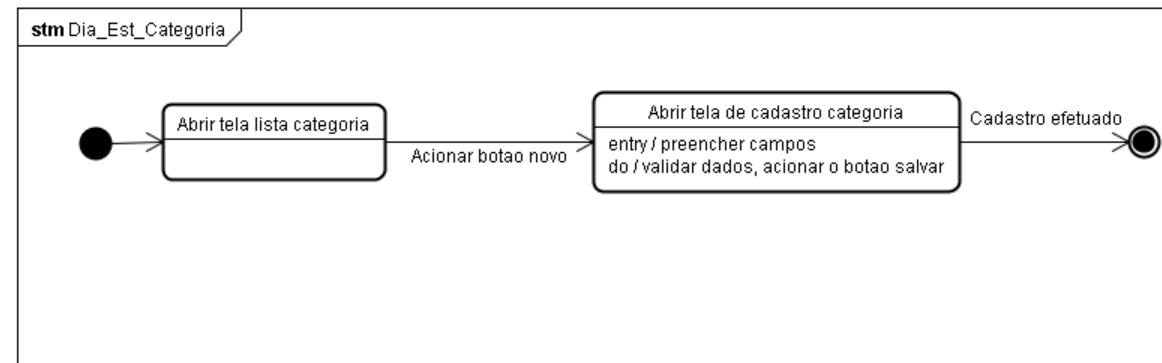


Diagrama de Transições de Estado – Pais

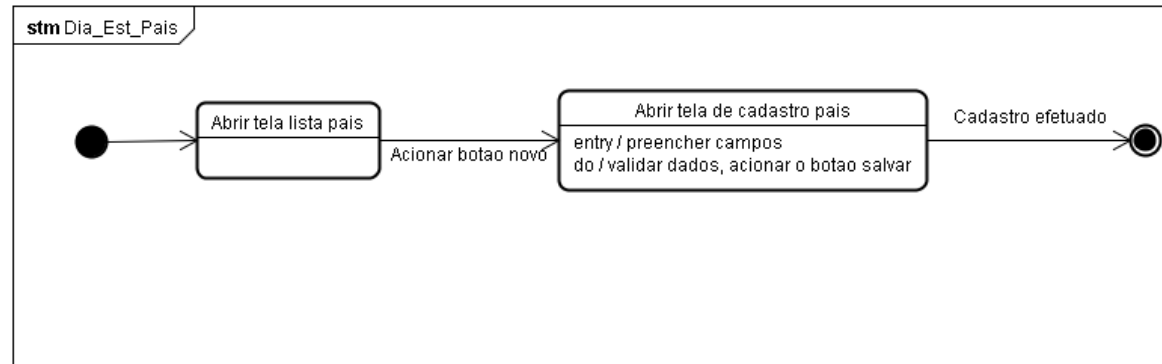


Diagrama de Transições de Estado – Estado

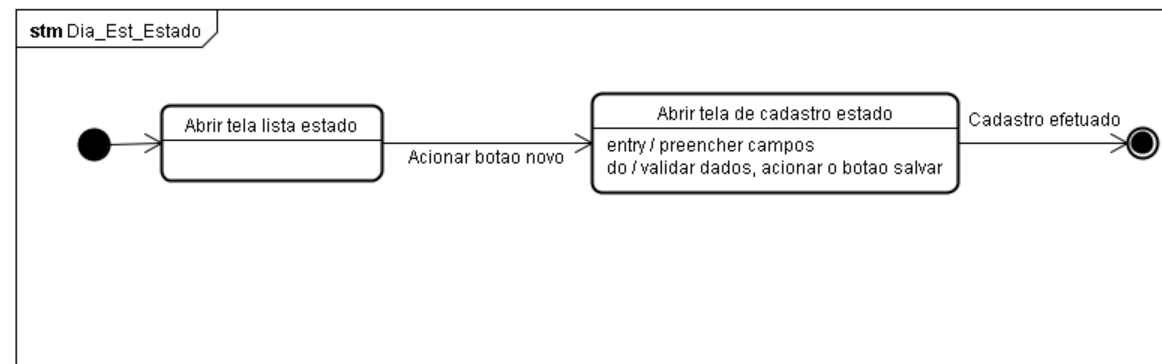


Diagrama de Transições de Estado – Cidade

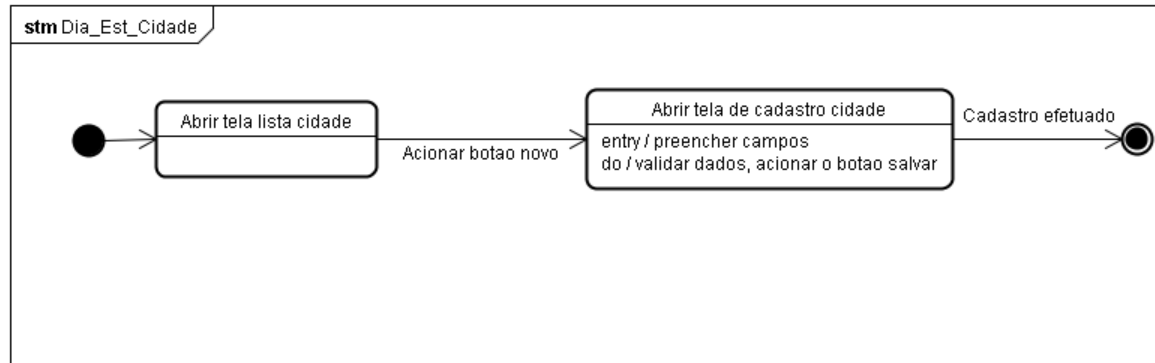


Diagrama de Transições de Estado – Usuário

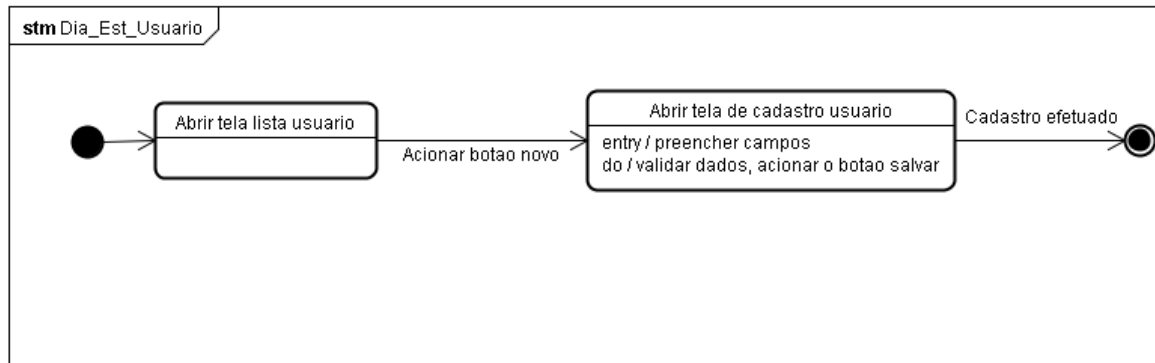


Diagrama de Transições de Estado – Perfil

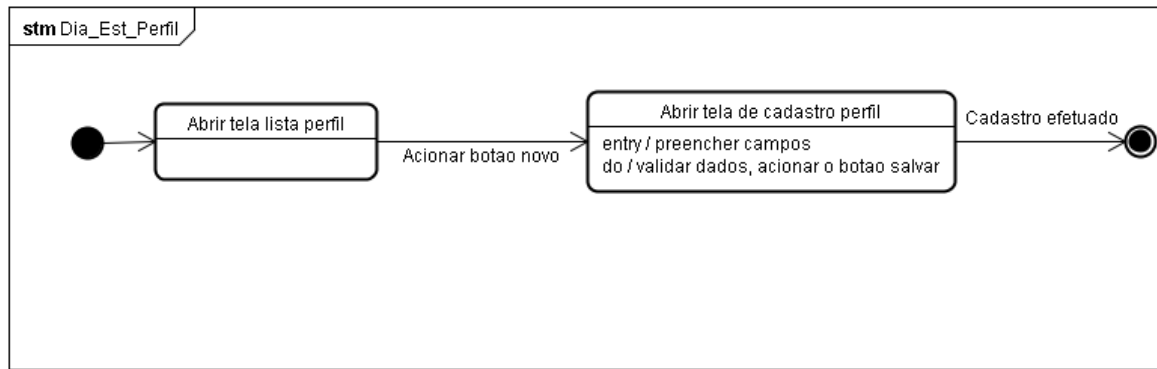


Diagrama de Transições de Estado – Atividade

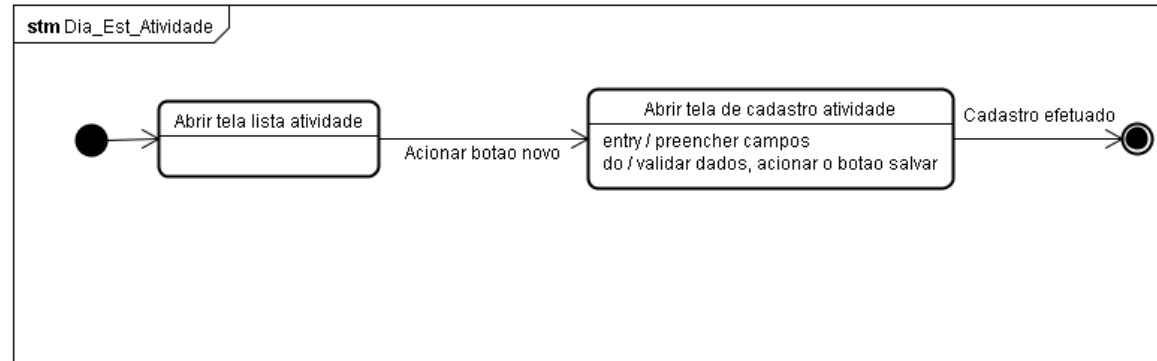


Diagrama de Transições de Estado – Login

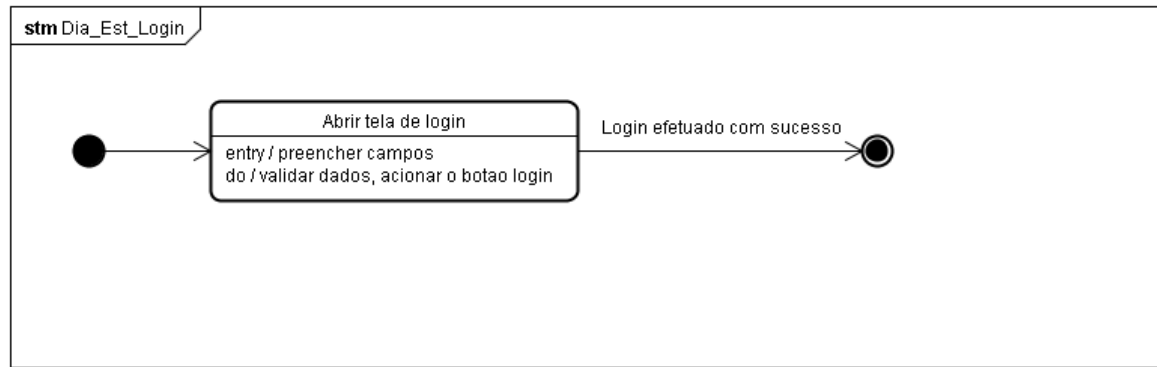


Diagrama de Transições de Estado – Android Empresa

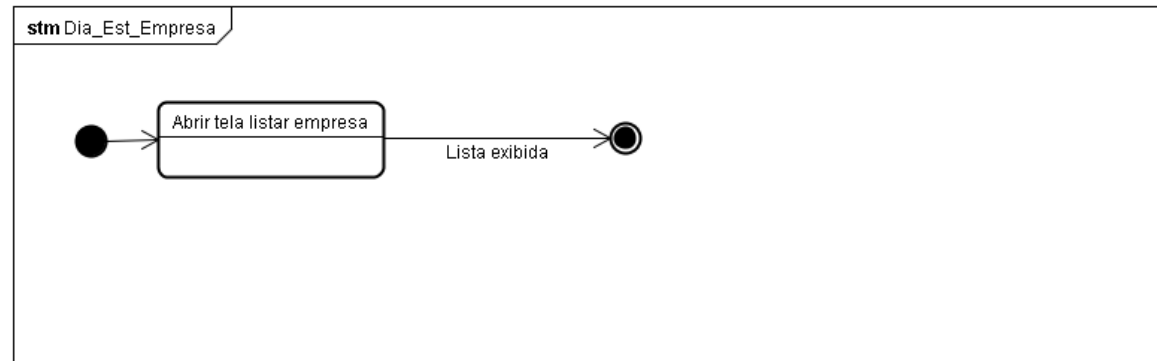


Diagrama de Transições de Estado – Android Produto



Diagrama de Transições de Estado – Android Detalhes do Produto



Plano de Teste

Versão 2.1

Histórico da Revisão

Data	Versão	Descrição	Autor
10/08/2013	1.0	Versão inicial	Maicon Alexander Illes Fang
11/12/2013	2.0	Versão secundaria	Maicon Alexander Illes Fang
25/02/2014	2.1	Ajustes e inserção de novas baterias.	Maicon Alexander Illes Fang

Plano de Teste – Empresa

Caso de Uso	UC001 – Manter Empresa		
Pré-condições	Ter acesso ao sistema.		
Elaborador	Maicon Alexander Illes Fang	Data de Elaboração	25/02/2014
Executor	Maicon Alexander Illes Fang	Data de Execução	15/05/2014

PLANEJAMENTO DO TESTE

Os testes serão feitos em 6 baterias.

Bateria 0 – Produzir todas as exceções previstas no caso de uso.

Bateria 1 – Testar as regras de negócio.

Bateria 2 – Executar teste case.

Bateria 3 – Testar banco de dados.

Bateria 4 – Teste de carga.

Bateria 5 – Testar segurança e controle de acesso.

Detalhamento das Baterias:

BATERIA 0

Produzir todas as exceções previstas do caso de uso.

BATERIA 1

Testar todas as regras de negócio referente ao caso de uso

BATERIA 2

Executar os casos de teste referente ao caso de uso.

BATERIA 3

Testar se os dados da empresa podem ser cadastrados, editados, consultados e removidos.

BATERIA 4

Verificar a resposta do sistema com 10, 75 e 150 usuários acessando o sistema.

BATERIA 5

Verificar que usuários não cadastrados não podem acessar informações restritas aos cadastrados.

Verificar que além do administrador, ninguém mais pode inserir, atualizar ou remover dados da empresa.

Verificar que os usuários do sistema podem acessar apenas as funcionalidades e dados associados ao seu próprio tipo de usuário.

Plano de Teste – Funcionário

Caso de Uso	UC002 – Manter Funcionário.		
Pré-condições	Ter acesso ao sistema.		
Elaborador	Maicon Alexander Illes Fang	Data de Elaboração	25/02/2014
Executor	Maicon Alexander Illes Fang	Data de Execução	15/05/2014

PLANEJAMENTO DO TESTE

Os testes serão feitos em 6 baterias.

Bateria 0 – Produzir todas as exceções previstas no caso de uso.

Bateria 1 – Testar as regras de negócio.

Bateria 2 – Executar teste case.

Bateria 3 – Testar banco de dados.

Bateria 4 – Teste de carga.

Bateria 5 – Testar segurança e controle de acesso.

Detalhamento das Baterias:

BATERIA 0

Produzir todas as exceções previstas do caso de uso.

BATERIA 1

Testar todas as regras de negócio referente ao caso de uso

BATERIA 2

Executar os casos de teste referente ao caso de uso.

BATERIA 3

Testar se os dados do funcionário podem ser cadastrados, editados, consultados e removidos.

BATERIA 4

Verificar a resposta do sistema com 10, 75 e 150 usuários acessando o sistema.

BATERIA 5

Verificar que usuários não cadastrados não podem acessar informações restritas aos cadastrados.

Verificar que além do administrador, ninguém mais pode inserir, atualizar ou remover dados do funcionário.

Verificar que os usuários do sistema podem acessar apenas as funcionalidades e dados associados ao seu próprio tipo de usuário.

Plano de Teste – Produto

Caso de Uso	UC003 – Manter Produto.		
Pré-condições	Ter acesso ao sistema.		
Elaborador	Maicon Alexander Illes Fang	Data de Elaboração	25/02/2014
Executor	Maicon Alexander Illes Fang	Data de Execução	15/05/2014

PLANEJAMENTO DO TESTE

Os testes serão feitos em 6 baterias.

Bateria 0 – Produzir todas as exceções previstas no caso de uso.

Bateria 1 – Testar as regras de negócio.

Bateria 2 – Executar teste case.

Bateria 3 – Testar banco de dados.

Bateria 4 – Teste de carga.

Bateria 5 – Testar segurança e controle de acesso.

Detalhamento das Baterias:

BATERIA 0

Produzir todas as exceções previstas do caso de uso.

BATERIA 1

Testar todas as regras de negócio referente ao caso de uso

BATERIA 2

Executar os casos de teste referente ao caso de uso.

BATERIA 3

Testar se os dados do produto podem ser cadastrados, editados, consultados e removidos.

BATERIA 4

Verificar a resposta do sistema com 10, 75 e 150 usuários acessando o sistema.

BATERIA 5

Verificar que usuários não cadastrados não podem acessar informações restritas aos cadastrados.

Verificar que além do administrador, ninguém mais pode inserir, atualizar ou remover dados do produto.

Verificar que os usuários do sistema podem acessar apenas as funcionalidades e dados associados ao seu próprio tipo de usuário.

Plano de Teste – Categoria

Caso de Uso	UC004 – Manter Categoria.		
Pré-condições	Ter acesso ao sistema.		
Elaborador	Maicon Alexander Illes Fang	Data de Elaboração	25/02/2014
Executor	Maicon Alexander Illes Fang	Data de Execução	15/05/2014

PLANEJAMENTO DO TESTE

Os testes serão feitos em 6 baterias.

Bateria 0 – Produzir todas as exceções previstas no caso de uso.

Bateria 1 – Testar as regras de negócio.

Bateria 2 – Executar teste case.

Bateria 3 – Testar banco de dados.

Bateria 4 – Teste de carga.

Bateria 5 – Testar segurança e controle de acesso.

Detalhamento das Baterias:

BATERIA 0

Produzir todas as exceções previstas do caso de uso.

BATERIA 1

Testar todas as regras de negócio referente ao caso de uso

BATERIA 2

Executar os casos de teste referente ao caso de uso.

BATERIA 3

Testar se os dados da categoria podem ser cadastrados, editados, consultados e removidos.

BATERIA 4

Verificar a resposta do sistema com 10, 75 e 150 usuários acessando o sistema.

BATERIA 5

Verificar que usuários não cadastrados não podem acessar informações restritas aos cadastrados.

Verificar que além do administrador, ninguém mais pode inserir, atualizar ou remover dados do categoria.

Verificar que os usuários do sistema podem acessar apenas as funcionalidades e dados associados ao seu próprio tipo de usuário.

Plano de Teste – Pais

Caso de Uso	UC005 – Manter Pais.		
Pré-condições	Ter acesso ao sistema.		
Elaborador	Maicon Alexander Illes Fang	Data de Elaboração	25/02/2014
Executor	Maicon Alexander Illes Fang	Data de Execução	15/05/2014

PLANEJAMENTO DO TESTE

Os testes serão feitos em 6 baterias.

Bateria 0 – Produzir todas as exceções previstas no caso de uso.

Bateria 1 – Testar as regras de negócio.

Bateria 2 – Executar teste case.

Bateria 3 – Testar banco de dados.

Bateria 4 – Teste de carga.

Bateria 5 – Testar segurança e controle de acesso.

Detalhamento das Baterias:

BATERIA 0

Produzir todas as exceções previstas do caso de uso.

BATERIA 1

Testar todas as regras de negócio referente ao caso de uso

BATERIA 2

Executar os casos de teste referente ao caso de uso.

BATERIA 3

Testar se os dados do pais podem ser cadastrados, editados, consultados e removidos.

BATERIA 4

Verificar a resposta do sistema com 10, 75 e 150 usuários acessando o sistema.

BATERIA 5

Verificar que usuários não cadastrados não podem acessar informações restritas aos cadastrados.

Verificar que além do administrador, ninguém mais pode inserir, atualizar ou remover dados do pais.

Verificar que os usuários do sistema podem acessar apenas as funcionalidades e dados associados ao seu próprio tipo de usuário.

Plano de Teste – Estado

Caso de Uso	UC006 – Manter Estado.
Pré-condições	Ter acesso ao sistema.

Elaborador	Maicon Alexander Illes Fang	Data de Elaboração	25/02/2014
Executor	Maicon Alexander Illes Fang	Data de Execução	15/05/2014

PLANEJAMENTO DO TESTE

Os testes serão feitos em 6 baterias.

Bateria 0 – Produzir todas as exceções previstas no caso de uso.

Bateria 1 – Testar as regras de negócio.

Bateria 2 – Executar teste case.

Bateria 3 – Testar banco de dados.

Bateria 4 – Teste de carga.

Bateria 5 – Testar segurança e controle de acesso.

Detalhamento das Baterias:

BATERIA 0

Produzir todas as exceções previstas do caso de uso.

BATERIA 1

Testar todas as regras de negócio referente ao caso de uso

BATERIA 2

Executar os casos de teste referente ao caso de uso.

BATERIA 3

Testar se os dados do estado podem ser cadastrados, editados, consultados e removidos.

BATERIA 4

Verificar a resposta do sistema com 10, 75 e 150 usuários acessando o sistema.

BATERIA 5

Verificar que usuários não cadastrados não podem acessar informações restritas aos cadastrados.

Verificar que além do administrador, ninguém mais pode inserir, atualizar ou remover dados do estado.

Verificar que os usuários do sistema podem acessar apenas as funcionalidades e dados associados ao seu próprio tipo de usuário.

Caso de Uso	UC007 – Manter Cidade.		
Pré-condições	Ter acesso ao sistema.		
Elaborador	Maicon Alexander Illes Fang	Data de Elaboração	25/02/2014
Executor	Maicon Alexander Illes Fang	Data de Execução	15/05/2014

PLANEJAMENTO DO TESTE

Os testes serão feitos em 6 baterias.

Bateria 0 – Produzir todas as exceções previstas no caso de uso.

Bateria 1 – Testar as regras de negócio.

Bateria 2 – Executar teste case.

Bateria 3 – Testar banco de dados.

Bateria 4 – Teste de carga.

Bateria 5 – Testar segurança e controle de acesso.

Detalhamento das Baterias:

BATERIA 0

Produzir todas as exceções previstas do caso de uso.

BATERIA 1

Testar todas as regras de negócio referente ao caso de uso

BATERIA 2

Executar os casos de teste referente ao caso de uso.

BATERIA 3

Testar se os dados da cidade podem ser cadastrados, editados, consultados e removidos.

BATERIA 4

Verificar a resposta do sistema com 10, 75 e 150 usuários acessando o sistema.

BATERIA 5

Verificar que usuários não cadastrados não podem acessar informações restritas aos cadastrados.

Verificar que além do administrador, ninguém mais pode inserir, atualizar ou remover dados do cidade.

Verificar que os usuários do sistema podem acessar apenas as funcionalidades e dados associados ao seu próprio tipo de usuário.

Plano de Teste – Usuário

Caso de Uso	UC008 – Manter Usuário.		
Pré-condições	Ter acesso ao sistema.		
Elaborador	Maicon Alexander Illes Fang	Data de Elaboração	25/02/2014
Executor	Maicon Alexander Illes Fang	Data de Execução	15/05/2014

PLANEJAMENTO DO TESTE

Os testes serão feitos em 6 baterias.

Bateria 0 – Produzir todas as exceções previstas no caso de uso.

Bateria 1 – Testar as regras de negócio.

Bateria 2 – Executar teste case.

Bateria 3 – Testar banco de dados.

Bateria 4 – Teste de carga.

Bateria 5 – Testar segurança e controle de acesso.

Detalhamento das Baterias:

BATERIA 0

Produzir todas as exceções previstas do caso de uso.

BATERIA 1

Testar todas as regras de negócio referente ao caso de uso

BATERIA 2

Executar os casos de teste referente ao caso de uso.

BATERIA 3

Testar se os dados do usuário podem ser cadastrados, editados, consultados e removidos.

BATERIA 4

Verificar a resposta do sistema com 10, 75 e 150 usuários acessando o sistema.

BATERIA 5

Verificar que usuários não cadastrados não podem acessar informações restritas aos cadastrados.

Verificar que além do administrador, ninguém mais pode inserir, atualizar ou remover dados do usuário.

Verificar que os usuários do sistema podem acessar apenas as funcionalidades e dados associados ao seu próprio tipo de usuário.

Plano de Teste – Perfil

Caso de Uso	UC009 – Manter Perfil.		
Pré-condições	Ter acesso ao sistema.		
Elaborador	Maicon Alexander Illes Fang	Data de Elaboração	25/02/2014
Executor	Maicon Alexander Illes Fang	Data de Execução	15/05/2014

PLANEJAMENTO DO TESTE

Os testes serão feitos em 6 baterias.

Bateria 0 – Produzir todas as exceções previstas no caso de uso.

Bateria 1 – Testar as regras de negócio.

Bateria 2 – Executar teste case.

Bateria 3 – Testar banco de dados.

Bateria 4 – Teste de carga.

Bateria 5 – Testar segurança e controle de acesso.

Detalhamento das Baterias:

BATERIA 0

Produzir todas as exceções previstas do caso de uso.

BATERIA 1

Testar todas as regras de negócio referente ao caso de uso

BATERIA 2

Executar os casos de teste referente ao caso de uso.

BATERIA 3

Testar se os dados do perfil podem ser cadastrados, editados, consultados e removidos.

BATERIA 4

Verificar a resposta do sistema com 10, 75 e 150 usuários acessando o sistema.

BATERIA 5

Verificar que usuários não cadastrados não podem acessar informações restritas aos cadastrados.

Verificar que além do administrador, ninguém mais pode inserir, atualizar ou remover dados do perfil.

Verificar que os usuários do sistema podem acessar apenas as funcionalidades e dados associados ao seu próprio tipo de usuário.

Plano de Teste – Atividade

Caso de Uso	UC010 – Manter Atividade.		
Pré-condições	Ter acesso ao sistema.		
Elaborador	Maicon Alexander Illes Fang	Data de Elaboração	25/02/2014
Executor	Maicon Alexander Illes Fang	Data de Execução	15/05/2014

PLANEJAMENTO DO TESTE

Os testes serão feitos em 6 baterias.

Bateria 0 – Produzir todas as exceções previstas no caso de uso.

Bateria 1 – Testar as regras de negócio.

Bateria 2 – Executar teste case.

Bateria 3 – Testar banco de dados.

Bateria 4 – Teste de carga.

Bateria 5 – Testar segurança e controle de acesso.

Detalhamento das Baterias:

BATERIA 0

Produzir todas as exceções previstas do caso de uso.

BATERIA 1

Testar todas as regras de negócio referente ao caso de uso

BATERIA 2

Executar os casos de teste referente ao caso de uso.

BATERIA 3

Testar se os dados da atividade podem ser cadastrados, editados, consultados e removidos.

BATERIA 4

Verificar a resposta do sistema com 10, 75 e 150 usuários acessando o sistema.

BATERIA 5

Verificar que usuários não cadastrados não podem acessar informações restritas aos cadastrados.

Verificar que além do administrador, ninguém mais pode inserir, atualizar ou remover dados da atividade.

Verificar que os usuários do sistema podem acessar apenas as funcionalidades e dados associados ao seu próprio tipo de usuário.

Plano de Teste – Login

Caso de Uso	UC011 – Manter Login.		
Pré-condições	Ter acesso ao sistema.		
Elaborador	Maicon Alexander Illes Fang	Data de Elaboração	25/02/2014
Executor	Maicon Alexander Illes Fang	Data de Execução	15/05/2014

PLANEJAMENTO DO TESTE

Os testes serão feitos em 4 baterias.

Bateria 0 – Produzir todas as exceções previstas no caso de uso.

Bateria 1 – Testar as regras de negócio.

Bateria 2 – Executar teste case.

Bateria 3 – Testar segurança e controle de acesso.

Detalhamento das Baterias:

BATERIA 0

Produzir todas as exceções previstas do caso de uso.

BATERIA 1

Testar todas as regras de negócio referente ao caso de uso

BATERIA 2

Executar os casos de teste referente ao caso de uso.

BATERIA 3

Verificar que usuários não cadastrados não podem acessar o sistema.

Plano de Teste – Empresa Android

Caso de Uso	UC500 – Exibir empresa/s na aplicação android.		
Pré-condições	Ter acesso ao sistema.		
Elaborador	Maicon Alexander Illes Fang	Data de Elaboração	25/02/2014
Executor	Maicon Alexander Illes Fang	Data de Execução	15/05/2014

PLANEJAMENTO DO TESTE

Os testes serão feitos em 6 baterias.

Bateria 0 – Produzir todas as exceções previstas no caso de uso.

Bateria 1 – Testar as regras de negócio.

Bateria 2 – Executar teste case.

Bateria 3 – Teste de carga.

Detalhamento das Baterias:

BATERIA 0

Produzir todas as exceções previstas do caso de uso.

BATERIA 1

Testar todas as regras de negócio referente ao caso de uso

BATERIA 2

Executar os casos de teste referente ao caso de uso.

BATERIA 3

Verificar a resposta do sistema com 10, 75 e 150 usuários acessando o sistema.

Plano de Teste – Produto Android

Caso de Uso	UC501 – Exibir produto/s na aplicação android.		
Pré-condições	Ter acesso ao sistema.		
Elaborador	Maicon Alexander Illes Fang	Data de Elaboração	25/02/2014
Executor	Maicon Alexander Illes Fang	Data de Execução	15/05/2014

PLANEJAMENTO DO TESTE

Os testes serão feitos em 6 baterias.

Bateria 0 – Produzir todas as exceções previstas no caso de uso.

Bateria 1 – Testar as regras de negócio.

Bateria 2 – Executar teste case.

Bateria 3 – Teste de carga.

Detalhamento das Baterias:

BATERIA 0

Produzir todas as exceções previstas do caso de uso.

BATERIA 1

Testar todas as regras de negócio referente ao caso de uso

BATERIA 2

Executar os casos de teste referente ao caso de uso.

BATERIA 3

Verificar a resposta do sistema com 10, 75 e 150 usuários acessando o sistema.

Plano de Teste – Detalhe do Produto Android

Caso de Uso	UC502 – Exibir detalhes de um produto na aplicação android.		
Pré-condições	Ter acesso ao sistema.		
Elaborador	Maicon Alexander Illes Fang	Data de Elaboração	25/02/2014
Executor	Maicon Alexander Illes Fang	Data de Execução	15/05/2014

PLANEJAMENTO DO TESTE

Os testes serão feitos em 6 baterias.

Bateria 0 – Produzir todas as exceções previstas no caso de uso.

Bateria 1 – Testar as regras de negócio.

Bateria 2 – Executar teste case.

Bateria 3 – Teste de carga.

Detalhamento das Baterias:

BATERIA 0

Produzir todas as exceções previstas do caso de uso.

BATERIA 1

Testar todas as regras de negócio referente ao caso de uso

BATERIA 2

Executar os casos de teste referente ao caso de uso.

BATERIA 3

Verificar a resposta do sistema com 10, 75 e 150 usuários acessando o sistema.

APÊNDICE P - 16 - Template Casos de Teste

Especificação do Caso de Teste Versão 2.1

Histórico da Revisão

Data	Versão	Descrição	Autor
10/08/2013	1.0	Versão inicial	Maicon Alexander Illes Fang
11/12/2013	2.0	Versão secundaria	Maicon Alexander Illes Fang
25/02/2014	2.1	Ajustes e inserção de novos casos de teste.	Maicon Alexander Illes Fang

Teste Case - UC001 – Manter Empresa

Cenário de Testes

Use Case	UC001 – Cadastrar Empresa	Descrição
Número	Cenário	
01	Cadastrar - Salvar registro sem preencher nenhum campo.	
02	Cadastrar - Validar cadastro com todos os campos preenchidos.	
03	Cadastrar - Validar CNPJ duplicado.	
04	Cadastrar - Validar campos com máximo de caracteres preenchidos.	
05	Cadastrar - Validar preenchimento de campos com caracteres inválidos.	
06	Cadastrar - Verificar se campo “Ativo” é carregado com opção “Sim”	
Use Case	UC001 – Editar Empresa	Descrição
Número	Cenário	
07	Editar – Validar consistência das informações.	Salvar um registro e editar o mesmo.
08	Editar – Validar campo em branco	Limpar todos os campos.
09	Editar - Validar CNPJ duplicado.	
10	Editar - Validar campos com máximo de caracteres preenchidos.	
11	Editar - Validar preenchimento de campos com caracteres inválidos.	
12	Editar - Verificar se campo “Ativo” é carregado com opção “Sim”	

Caso de Uso	UC001 – Cadastrar Empresa		
Pré-condições	Ter uma cidade e atividade cadastrada no sistema. O sistema deverá ter executado o caso de uso Novo ou Alterar O usuário deve possuir permissões para Salvar os dados no sistema.		
Elaborador	Maicon Alexander Illes Fang	Data de Elaboração	12/08/2013

Executor			Maicon Alexander Illes Fang			Data de Execução		15/05/2014
N.º	Nome	Pré- condições	Passo	Entrada	Ação	Resultado Esperado	Status	
01	Cadastrar - Salvar registro sem preencher nenhum campo.	Os combo box cidade, atividade deve estar carregado. O botão salvar estar habilitado.	01	Não preencher nenhum campo.	Clicar no botão <i>Salvar</i> .	O sistema exibe a Mensagem “Existem campos obrigatórios que devem ser preenchidos. Por favor, preencha os campos que estão destacados.”. O sistema exibe na tela, em cor vermelha, o rótulo do campo que possui “*” e que não foi preenchido.		
02	Cadastrar - Validar cadastro com todos os campos preenchidos.	Os combo box cidade, atividade deve estar carregado. Ter um CNPJ, IE, CEP válido O botão salvar estar habilitado.	01	Preencher todos os campos obrigatórios.	Clicar no botão <i>Salvar</i> .	O sistema grava as informações no sistema e exibe a mensagem “Registro salvo com Sucesso” e permanece na tela de cadastro de empresa. O teste case é finalizado.		

03	Cadastrar - Validar CNPJ duplicado.	Os combo box cidade, atividade deve estar carregado. Ter um IE, CEP válido. Ter um CNPJ duplicado. O botão salvar estar habilitado.	01	No campo “CNPJ” preencher com um CNPJ já existente no sistema. Preencher os demais campos obrigatórios.	Clicar no botão <i>Salvar</i> .	O sistema exibe a mensagem “Não é permitido salvar o registro, CNPJ duplicado”, limpa o campo “CNPJ” e permanece na tela de cadastro de empresa. O teste case é finalizado.	
04	Cadastrar - Validar campos com máximo de caracteres preenchidos.	Os combo box cidade, atividade deve estar carregado. Ter um CNPJ, IE, CEP válido. O botão salvar estar habilitado.	01	Preencher todos os campos com máximo de caracteres.	Clicar no botão <i>Salvar</i> .	O sistema grava as informações no sistema e exibe a mensagem “Registro salvo com Sucesso” e permanece na tela de cadastro de empresa. O teste case é finalizado.	
05	Cadastrar - Validar preenchimento de campos com caracteres inválidos.	Os combo box cidade, atividade deve estar carregado. Ter um CNPJ, IE, CEP válido. O botão salvar estar habilitado.	01	Preencher todos os campos com caracteres inválidos.	Clicar no botão <i>Salvar</i> .	O sistema não permite preencher os campos com caracteres inválidos. O teste case é finalizado.	

06	Cadastrar - Verificar se campo "Ativo" é carregado com opção "Sim"	Os combo box cidade, atividade deve estar carregado. Ter um CNPJ, IE, CEP válido. O botão salvar estar habilitado.	01	Não se aplica.	Visualizar campo "Ativo"	O sistema exibe o campo com opção "Sim" marcada. O teste case é finalizado.	
07	Editar – Validar alterações das informações.	Editar um registro salvo.	01	Alterar todos os campos habilitados.	Pressionar o botão "Editar"	O sistema edita as informações no sistema e exibe a mensagem "Registro editado com Sucesso" e permanece na tela de cadastro de empresa. O teste case é finalizado.	
08	Editar – Validar campo em branco	Editar um registro salvo.	01	Limpar todos os campos habilitados.	Pressionar o botão "Editar"	O sistema não permite ação e exibe a mensagem "Campos Obrigatórios não foram preenchidos".	

09	Editar - Validar CNPJ duplicado.	Os combo box cidade, atividade deve estar carregado. Ter um IE, CEP válido. Ter um CNPJ duplicado. O botão editar estar habilitado.	01	No campo “CNPJ” preencher com um CNPJ já existente no sistema. Preencher os demais campos obrigatórios.	Clicar no botão “Editar”.	O sistema exibe a mensagem “Não é permitido editar o registro, CNPJ duplicado”, limpa o campo “CNPJ” e permanece na tela de edição da empresa. O teste case é finalizado.	
10	Editar - Validar campos com máximo de caracteres preenchidos.	Os combo box cidade, atividade deve estar carregado. Ter um CNPJ, IE, CEP válido. O botão editar estar habilitado.	01	Preencher todos os campos com máximo de caracteres.	Clicar no botão “ <i>Editar</i> ”.	O sistema grava as informações no sistema e exibe a mensagem “Registro editado com Sucesso” e permanece na tela de edição da empresa. O teste case é finalizado.	
11	Editar - Validar preenchimento de campos com caracteres inválidos.	Os combo box cidade, atividade deve estar carregado. Ter um CNPJ, IE, CEP válido. O botão salvar estar habilitado.	01	Preencher todos os campos com caracteres inválidos.	Clicar no botão “ <i>Editar</i> ”.	O sistema não permite preencher os campos com caracteres inválidos. O teste case é finalizado.	

12	Editar - Verificar se campo "Ativo" é carregado com opção "Sim"	Os combo box cidade, atividade deve estar carregado. Ter um CNPJ, IE, CEP válido. O botão salvar estar habilitado.	01	Não se aplica.	Visualizar campo "Ativo"	O sistema exibe o campo com opção "Sim" marcada. O teste case é finalizado.	
----	--	--	----	-------------------	-----------------------------	---	--

Teste Case - UC002 – Manter Funcionário

Cenário de Teste

Use Case	UC002 – Cadastrar Funcionário	Descrição
Número	Cenário	
01	Cadastrar - Salvar registro sem preencher nenhum campo.	
02	Cadastrar - Validar cadastro com todos os campos preenchidos.	
03	Cadastrar - Validar CPF duplicado.	
04	Cadastrar - Validar campos com máximo de caracteres preenchidos.	
05	Cadastrar - Validar preenchimento de campos com caracteres inválidos.	
06	Cadastrar - Verificar se campo "Ativo" é carregado com opção "Sim"	

Use Case	UC002 – Editar Funcionário	Descrição
Número	Cenário	
07	Editar – Validar consistência das informações.	Salvar um registro e editar o mesmo.
08	Editar – Validar campo em branco	Limpar todos os campos.
09	Editar - Validar CPF duplicado.	
10	Editar - Validar campos com máximo de caracteres preenchidos.	
11	Editar - Validar preenchimento de campos com caracteres inválidos.	
12	Editar - Verificar se campo “Ativo” é carregado com opção “Sim”	

Caso de Uso			UC002 – Cadastrar Funcionário				
Pré-condições			Ter uma cidade, empresa cadastrada no sistema. O sistema deverá ter executado o caso de uso Novo ou Alterar O usuário deve possuir permissões para Salvar os dados no sistema.				
Elaborador			Maicon Alexander Illes Fang		Data de Elaboração		13/08/2013
Executor			Maicon Alexander Illes Fang		Data de Execução		15/05/2014
N.º	Nome	Pré- condições	Passo	Entrada	Ação	Resultado Esperado	Status

01	Cadastrar - Salvar registro sem preencher nenhum campo.	Os combo box cidade, empresa deve estar carregado. O botão Cadastrar estar habilitado.	01	Não preencher nenhum campo.	Clicar no botão <i>Cadastrar</i> .	O sistema exibe a Mensagem "Existem campos obrigatórios que devem ser preenchidos. Por favor, preencha os campos que estão destacados.". O sistema exibe na tela, em cor vermelha, o rótulo do campo que possui "*" e que não foi preenchido.	
02	Cadastrar - Validar cadastro com todos os campos preenchidos.	Os combo box cidade, empresa deve estar carregado. O botão Cadastrar estar habilitado.	01	Preencher todos os campos obrigatórios.	Clicar no botão <i>Cadastrar</i> .	O sistema grava as informações no sistema e exibe a mensagem "Registro salvo com Sucesso" e permanece na tela de cadastro de funcionário. O teste case é finalizado.	

03	Cadastrar - Validar CPF duplicado.	Os combo box cidade, empresa deve estar carregado. O botão Cadastrar estar habilitado. Ter um CPF duplicado.	01	No campo “CPF” preencher com um CPF já existente no sistema. Preencher os demais campos obrigatórios.	Clicar no botão <i>Cadastrar</i> .	O sistema exibe a mensagem “Não é permitido salvar o registro, CPF duplicado”, limpa o campo “CPF” e permanece na tela de cadastro de funcionário. O teste case é finalizado.	
04	Cadastrar - Validar campos com máximo de caracteres preenchidos.	Os combo box cidade, empresa deve estar carregado. O botão Cadastrar estar habilitado.	01	Preencher todos os campos com máximo de caracteres.	Clicar no botão <i>Cadastrar</i> .	O sistema grava as informações no sistema e exibe a mensagem “Registro salvo com Sucesso” e permanece na tela de cadastro de funcionário. O teste case é finalizado.	
05	Cadastrar - Validar preenchimento de campos com caracteres inválidos.	Os combo box cidade, empresa deve estar carregado. O botão Cadastrar estar habilitado.	01	Preencher todos os campos com caracteres inválidos.	Clicar no botão <i>Cadastrar</i> .	O sistema não permite preencher os campos com caracteres inválidos. O teste case é finalizado.	

06	Cadastrar - Verificar se campo "Ativo" é carregado com opção "Sim"	Os combo box cidade, empresa deve estar carregado. O botão Cadastrar estar habilitado.	01	Não se aplica.	Visualizar campo "Ativo"	O sistema exibe o campo com opção "Sim" marcada. O teste case é finalizado.	
07	Editar – Validar alterações das informações.	Editar um registro salvo.	01	Alterar todos os campos habilitados.	Pressionar o botão "Editar"	O sistema edita as informações no sistema e exibe a mensagem "Registro editado com Sucesso" e permanece na tela de cadastro de funcionário. O teste case é finalizado.	
08	Editar – Validar campo em branco	Editar um registro salvo.	01	Limpar todos os campos habilitados.	Pressionar o botão "Editar"	O sistema não permite ação e exibe a mensagem "Campos Obrigatórios não foram preenchidos".	
09	Editar - Validar CPF duplicado.	Os combo box cidade, empresa deve estar carregado. O botão Cadastrar estar habilitado. Ter um CPF duplicado	01	No campo "CPF" preencher com um "CPF" já existente no sistema. Preencher os demais campos obrigatórios.	Clicar no botão "Editar".	O sistema exibe a mensagem "Não é permitido editar o registro, CNPJ duplicado", limpa o campo "CNPJ" e permanece na tela de edição do funcionário. O teste case é finalizado.	

10	Editar - Validar campos com máximo de caracteres preenchidos.	Os combo box cidade, empresa deve estar carregado. O botão Cadastrar estar habilitado.	01	Preencher todos os campos com máximo de caracteres.	Clicar no botão “ <i>Editar</i> ”.	O sistema grava as informações no sistema e exibe a mensagem “Registro editado com Sucesso” e permanece na tela de edição do funcionário. O teste case é finalizado.	
11	Editar - Validar preenchimento de campos com caracteres inválidos.	Os combo box cidade, empresa deve estar carregado. O botão Cadastrar estar habilitado.	01	Preencher todos os campos com caracteres inválidos.	Clicar no botão “ <i>Editar</i> ”.	O sistema não permite preencher os campos com caracteres inválidos. O teste case é finalizado.	
12	Editar - Verificar se campo “Ativo” é carregado com opção “Sim”	Os combo box cidade, empresa deve estar carregado. O botão Cadastrar estar habilitado.	01	Não se aplica.	Visualizar campo “Ativo”	O sistema exibe o campo com opção “Sim” marcada. O teste case é finalizado.	

Teste Case - UC003 – Manter Produto

Casos de Teste

Use Case	UC003 – Cadastrar Produto	Descrição
Número	Cenário	
01	Cadastrar - Salvar registro sem preencher nenhum campo.	
02	Cadastrar - Validar cadastro com todos os campos preenchidos.	
04	Cadastrar - Validar campos com máximo de caracteres preenchidos.	
05	Cadastrar - Validar preenchimento de campos com caracteres inválidos.	
06	Cadastrar - Verificar se campo “Ativo” é carregado com opção “Sim”	
Use Case	UC003 – Editar Produto	Descrição
Número	Cenário	
07	Editar – Validar consistência das informações.	Salvar um registro e editar o mesmo.
08	Editar – Validar campo em branco	Limpar todos os campos.
10	Editar - Validar campos com máximo de caracteres preenchidos.	
11	Editar - Validar preenchimento de campos com caracteres inválidos.	
12	Editar - Verificar se campo “Ativo” é carregado com opção “Sim”	

Caso de Uso	UC003 – Cadastrar Produto		
Pré-condições	Ter um Funcionário e Empresa e Categoria cadastrada no sistema. O sistema deverá ter executado o caso de uso Novo ou Alterar. O usuário deve possuir permissões para Salvar os dados no sistema.		
Elaborador	Maicon Alexander Illes Fang	Data de Elaboração	13/08/2013
Executor	Maicon Alexander Illes Fang	Data de Execução	15/05/2014

N.º	Nome	Pré-condições	Passo	Entrada	Ação	Resultado Esperado	Status
01	Cadastrar - Salvar registro sem preencher nenhum campo.	Os combo box Funcionário e Empresa e Categoria deve estar carregado. O botão salvar estar habilitado.	01	Não preencher nenhum campo.	Clicar no botão <i>Cadastrar</i> .	O sistema exibe a Mensagem “Existem campos obrigatórios que devem ser preenchidos. Por favor, preencha os campos que estão destacados.”. O sistema exibe na tela, em cor vermelha, o rótulo do campo que possui “*” e que não foi preenchido. O teste case é finalizado.	
02	Cadastrar - Validar cadastro com todos os campos preenchidos.	Os combo box Funcionário e Empresa e Categoria deve estar carregado. O botão salvar estar habilitado.	01	Preencher todos os campos obrigatórios.	Clicar no botão <i>Cadastrar</i> .	O sistema grava as informações no sistema e exibe a mensagem “Registro salvo com Sucesso” e permanece na tela de cadastro de produto. O teste case é finalizado.	

03	Cadastrar - Validar campos com máximo de caracteres preenchidos.	Os combo box Funcionário e Empresa e Categoria deve estar carregado. O botão salvar estar habilitado.	01	Preencher todos os campos com máximo de caracteres.	Clicar no botão <i>Cadastrar</i> .	O sistema grava as informações no sistema e exibe a mensagem "Registro salvo com Sucesso" e permanece na tela de cadastro de produto. O teste case é finalizado.	
04	Cadastrar - Validar preenchimento de campos com caracteres inválidos.	Os combo box Funcionário e Empresa e Categoria deve estar carregado. O botão salvar estar habilitado.	01	Preencher todos os campos com caracteres inválidos.	Clicar no botão <i>Cadastrar</i> .	O sistema não permite preencher os campos com caracteres inválidos. O teste case é finalizado.	
05	Cadastrar - Verificar se campo "Ativo" é carregado com opção "Sim"	Os combo box Funcionário e Empresa e Categoria deve estar carregado. O botão salvar estar habilitado.	01	Não se aplica.	Visualizar campo "Ativo"	O sistema exibe o campo com opção "Sim" marcada. O teste case é finalizado.	

06	Editar – Validar alterações das informações.	Editar um registro salvo.	01	Alterar todos os campos habilitados.	Pressionar o botão “Editar”	O sistema edita as informações no sistema e exibe a mensagem “Registro editado com Sucesso” e permanece na tela de cadastro de produto. O teste case é finalizado.	
07	Editar – Validar campo em branco	Editar um registro salvo.	01	Limpar todos os campos habilitados.	Pressionar o botão “Editar”	O sistema não permite ação e exibe a mensagem “Campos Obrigatórios não foram preenchidos”.	
08	Editar - Validar campos com máximo de caracteres preenchidos.	Os combo box Funcionário e Empresa e Categoria deve estar carregado. O botão editar estar habilitado.	01	Preencher todos os campos com máximo de caracteres.	Clicar no botão “ <i>Editar</i> ”.	O sistema grava as informações no sistema e exibe a mensagem “Registro editado com Sucesso” e permanece na tela de edição de produto. O teste case é finalizado.	

09	Editar - Validar preenchimento de campos com caracteres inválidos.	Os combo box Funcionário e Empresa e Categoria deve estar carregado. O botão editar estar habilitado.	01	Preencher todos os campos com caracteres inválidos.	Clicar no botão “ <i>Editar</i> ”.	O sistema não permite preencher os campos com caracteres inválidos. O teste case é finalizado.	
10	Editar - Verificar se campo “Ativo” é carregado com opção “Sim”	Os combo box Funcionário e Empresa e Categoria deve estar carregado. O botão editar estar habilitado.	01	Não se aplica.	Visualizar campo “Ativo”	O sistema exibe o campo com opção “Sim” marcada. O teste case é finalizado.	

Teste Case - UC004 – Manter Categoria

Casos de Teste

Use Case	UC004 – Cadastrar Categoria	Descrição
Número	Cenário	
01	Cadastrar - Salvar registro sem preencher nenhum campo.	
02	Cadastrar - Validar cadastro com todos os campos preenchidos.	
03	Cadastrar - Validar campos com máximo de caracteres preenchidos.	

04	Cadastrar - Validar preenchimento de campos com caracteres inválidos.	
Use Case	UC004 – Editar Categoria	Descrição
Número	Cenário	
05	Editar – Validar alterações das informações.	Salvar um registro e editar o mesmo.
06	Editar – Validar campo em branco	Limpar todos os campos.
07	Editar - Validar campos com máximo de caracteres preenchidos.	
08	Editar - Validar preenchimento de campos com caracteres inválidos.	

Caso de Uso		UC004 – Cadastrar Categoria					
Pré-condições		Ter acesso ao sistema. O sistema deverá ter executado o caso de uso Novo ou Alterar. O usuário deve possuir permissões para Salvar os dados no sistema.					
Elaborador		Maicon Alexander Illes Fang			Data de Elaboração		13/08/2013
Executor		Maicon Alexander Illes Fang			Data de Execução		15/05/2014
N.º	Nome	Pré-condições	Passo	Entrada	Ação	Resultado Esperado	Status
01	Cadastrar - Salvar registro sem preencher nenhum campo.	O botão salvar estar habilitado.	01	Não preencher nenhum campo.	Clicar no botão <i>Cadastrar</i> .	O sistema exibe a Mensagem "Não podem haver campos vazios". O teste case é finalizado.	

02	Cadastrar - Validar cadastro com todos os campos preenchidos.	O botão salvar estar habilitado.	01	Preencher todos os campos obrigatórios.	Clicar no botão <i>Cadastrar</i> .	O sistema grava as informações no sistema e exibe a mensagem "Registro salvo com Sucesso" e permanece na mesma tela. O teste case é finalizado.	
03	Cadastrar - Validar campos com máximo de caracteres preenchidos.	O botão salvar estar habilitado.	01	Preencher todos os campos com máximo de caracteres.	Clicar no botão <i>Cadastrar</i> .	O sistema grava as informações no sistema e exibe a mensagem "Registro salvo com Sucesso" e permanece na mesma tela. O teste case é finalizado.	
04	Cadastrar - Validar preenchimento de campos com caracteres inválidos.	O botão salvar estar habilitado.	01	Preencher todos os campos com caracteres inválidos.	Clicar no botão <i>Cadastrar</i> .	O sistema não permite preencher os campos com caracteres inválidos. O teste case é finalizado.	
05	Editar – Validar alterações das informações.	Editar um registro salvo.	01	Alterar todos os campos habilitados.	Pressionar o botão "Editar"	O sistema edita as informações no sistema e exibe a mensagem "Registro editado com Sucesso" e permanece na mesma tela. O teste case é finalizado.	

06	Editar – Validar campo em branco	Editar um registro salvo.	01	Limpar todos os campos habilitados.	Pressionar o botão “Editar”	O sistema não permite ação e exibe a mensagem “Não podem haver campos vazios”. O teste case é finalizado	
07	Editar - Validar campos com máximo de caracteres preenchidos.	O botão editar estar habilitado.	01	Preencher todos os campos com máximo de caracteres.	Clicar no botão “ <i>Editar</i> ”.	O sistema grava as informações e exibe a mensagem “Alteração realizada com sucesso” e permanece na mesma tela. O teste case é finalizado.	
08	Editar - Validar preenchimento de campos com caracteres inválidos.	O botão editar estar habilitado.	01	Preencher todos os campos com caracteres inválidos.	Clicar no botão “ <i>Editar</i> ”.	O sistema não permite preencher os campos com caracteres inválidos. O teste case é finalizado.	

Teste Case - UC005 – Manter Pais

Casos de Teste

Use Case	UC005 – Cadastrar Pais	Descrição
Número	Cenário	
01	Cadastrar - Salvar registro sem preencher nenhum campo.	

02	Cadastrar - Validar cadastro com todos os campos preenchidos.	
03	Cadastrar - Validar campos com máximo de caracteres preenchidos.	
04	Cadastrar - Validar preenchimento de campos com caracteres inválidos.	
Use Case	UC005 – Editar Pais	Descrição
Número	Cenário	
05	Editar – Validar alterações das informações.	Salvar um registro e editar o mesmo.
06	Editar – Validar campo em branco	Limpar todos os campos.
07	Editar - Validar campos com máximo de caracteres preenchidos.	
08	Editar - Validar preenchimento de campos com caracteres inválidos.	

Caso de Uso			UC005 – Cadastrar Pais				
Pré-condições			Ter acesso ao sistema. O sistema deverá ter executado o caso de uso Novo ou Alterar. O usuário deve possuir permissões para Salvar os dados no sistema.				
Elaborador			Maicon Alexander Illes Fang		Data de Elaboração		13/08/2013
Executor			Maicon Alexander Illes Fang		Data de Execução		15/05/2014
N.º	Nome	Pré-condições	Passo	Entrada	Ação	Resultado Esperado	Status
01	Cadastrar - Salvar registro sem preencher nenhum campo.	O botão salvar estar habilitado.	01	Não preencher nenhum campo.	Clicar no botão <i>Cadastrar</i> .	O sistema exibe a Mensagem "Não podem haver campos vazios". O teste case é finalizado.	

02	Cadastrar - Validar cadastro com todos os campos preenchidos.	O botão salvar estar habilitado.	01	Preencher todos os campos obrigatórios.	Clicar no botão <i>Cadastrar</i> .	O sistema grava as informações no sistema e exibe a mensagem "Registro salvo com Sucesso" e permanece na mesma tela. O teste case é finalizado.	
03	Cadastrar - Validar campos com máximo de caracteres preenchidos.	O botão salvar estar habilitado.	01	Preencher todos os campos com máximo de caracteres.	Clicar no botão <i>Cadastrar</i> .	O sistema grava as informações no sistema e exibe a mensagem "Registro salvo com Sucesso" e permanece na mesma tela. O teste case é finalizado.	
04	Cadastrar - Validar preenchimento de campos com caracteres inválidos.	O botão salvar estar habilitado.	01	Preencher todos os campos com caracteres inválidos.	Clicar no botão <i>Cadastrar</i> .	O sistema não permite preencher os campos com caracteres inválidos. O teste case é finalizado.	
05	Editar – Validar alterações das informações.	Editar um registro salvo.	01	Alterar todos os campos habilitados.	Pressionar o botão "Editar"	O sistema edita as informações no sistema e exibe a mensagem "Registro editado com Sucesso" e permanece na mesma tela. O teste case é finalizado.	

06	Editar – Validar campo em branco	Editar um registro salvo.	01	Limpar todos os campos habilitados.	Pressionar o botão “Editar”	O sistema não permite ação e exibe a mensagem “Não podem haver campos vazios”. O teste case é finalizado	
07	Editar - Validar campos com máximo de caracteres preenchidos.	O botão editar estar habilitado.	01	Preencher todos os campos com máximo de caracteres.	Clicar no botão “ <i>Editar</i> ”.	O sistema grava as informações e exibe a mensagem “Alteração realizada com sucesso” e permanece na mesma tela. O teste case é finalizado.	
08	Editar - Validar preenchimento de campos com caracteres inválidos.	O botão editar estar habilitado.	01	Preencher todos os campos com caracteres inválidos.	Clicar no botão “ <i>Editar</i> ”.	O sistema não permite preencher os campos com caracteres inválidos. O teste case é finalizado.	

Teste Case - UC006 – Manter Estado

Casos de Teste

Use Case	UC006 – Cadastrar Estado	Descrição
Número	Cenário	
01	Cadastrar - Salvar registro sem preencher nenhum campo.	

02	Cadastrar - Validar cadastro com todos os campos preenchidos.	
03	Cadastrar - Validar campos com máximo de caracteres preenchidos.	
04	Cadastrar - Validar preenchimento de campos com caracteres inválidos.	
Use Case	UC006 – Editar Estado	Descrição
Número	Cenário	
05	Editar – Validar alterações das informações.	Salvar um registro e editar o mesmo.
06	Editar – Validar campo em branco	Limpar todos os campos.
07	Editar - Validar campos com máximo de caracteres preenchidos.	
08	Editar - Validar preenchimento de campos com caracteres inválidos.	

Caso de Uso		UC006 – Cadastrar Estado					
Pré-condições		Ter acesso ao sistema. O sistema deverá ter executado o caso de uso Novo ou Alterar. O usuário deve possuir permissões para Salvar os dados no sistema.					
Elaborador		Maicon Alexander Illes Fang			Data de Elaboração		13/08/2013
Executor		Maicon Alexander Illes Fang			Data de Execução		15/05/2014
N.º	Nome	Pré-condições	Passo	Entrada	Ação	Resultado Esperado	Status
01	Cadastrar - Salvar registro sem preencher nenhum campo.	Os combobox cidade deve estar carregado. O botão salvar estar habilitado.	01	Não preencher nenhum campo.	Clicar no botão <i>Cadastrar</i> .	O sistema exibe a Mensagem “Não podem haver campos vazios”. O teste case é finalizado.	

02	Cadastrar - Validar cadastro com todos os campos preenchidos.	Os combobox cidade deve estar carregado. O botão salvar estar habilitado.	01	Preencher todos os campos obrigatórios.	Clicar no botão <i>Cadastrar</i> .	O sistema grava as informações no sistema e exibe a mensagem "Registro salvo com Sucesso" e permanece na mesma tela. O teste case é finalizado.	
03	Cadastrar - Validar campos com máximo de caracteres preenchidos.	Os combobox cidade deve estar carregado. O botão salvar estar habilitado.	01	Preencher todos os campos com máximo de caracteres.	Clicar no botão <i>Cadastrar</i> .	O sistema grava as informações no sistema e exibe a mensagem "Registro salvo com Sucesso" e permanece na mesma tela. O teste case é finalizado.	
04	Cadastrar - Validar preenchimento de campos com caracteres inválidos.	O botão salvar estar habilitado.	01	Preencher todos os campos com caracteres inválidos.	Clicar no botão <i>Cadastrar</i> .	O sistema não permite preencher os campos com caracteres inválidos. O teste case é finalizado.	
05	Editar – Validar alterações das informações.	Os combobox cidade deve estar carregado. Editar um registro salvo.	01	Alterar todos os campos habilitados.	Pressionar o botão "Editar"	O sistema edita as informações no sistema e exibe a mensagem "Registro editado com Sucesso" e permanece na mesma tela. O teste case é finalizado.	

06	Editar – Validar campo em branco	Editar um registro salvo.	01	Limpar todos os campos habilitados.	Pressionar o botão “Editar”	O sistema não permite ação e exibe a mensagem “Não podem haver campos vazios”. O teste case é finalizado	
07	Editar - Validar campos com máximo de caracteres preenchidos.	O Os combobox cidade deve estar carregado. O botão editar estar habilitado.	01	Preencher todos os campos com máximo de caracteres.	Clicar no botão “ <i>Editar</i> ”.	O sistema grava as informações e exibe a mensagem “Alteração realizada com sucesso” e permanece na mesma tela. O teste case é finalizado.	
08	Editar - Validar preenchimento de campos com caracteres inválidos.	O botão editar estar habilitado.	01	Preencher todos os campos com caracteres inválidos.	Clicar no botão “ <i>Editar</i> ”.	O sistema não permite preencher os campos com caracteres inválidos. O teste case é finalizado.	

Teste Case - UC007 – Manter Cidade

Casos de Teste

Use Case	UC007 – Cadastrar Cidade	Descrição
Número	Cenário	
01	Cadastrar - Salvar registro sem preencher nenhum campo.	

02	Cadastrar - Validar cadastro com todos os campos preenchidos.	
03	Cadastrar - Validar campos com máximo de caracteres preenchidos.	
04	Cadastrar - Validar preenchimento de campos com caracteres inválidos.	
Use Case	UC007 – Editar Cidade	Descrição
Número	Cenário	
05	Editar – Validar alterações das informações.	Salvar um registro e editar o mesmo.
06	Editar – Validar campo em branco	Limpar todos os campos.
07	Editar - Validar campos com máximo de caracteres preenchidos.	
08	Editar - Validar preenchimento de campos com caracteres inválidos.	

Caso de Uso		UC007 – Cadastrar Cidade					
Pré-condições		Ter acesso ao sistema. O sistema deverá ter executado o caso de uso Novo ou Alterar. O usuário deve possuir permissões para Salvar os dados no sistema.					
Elaborador		Maicon Alexander Illes Fang			Data de Elaboração		13/08/2013
Executor		Maicon Alexander Illes Fang			Data de Execução		15/05/2014
N.º	Nome	Pré-condições	Passo	Entrada	Ação	Resultado Esperado	Status
01	Cadastrar - Salvar registro sem preencher nenhum campo.	Os combobox estado deve estar carregado. O botão salvar estar habilitado.	01	Não preencher nenhum campo.	Clicar no botão <i>Cadastrar</i> .	O sistema exibe a Mensagem “Não podem haver campos vazios”. O teste case é finalizado.	

02	Cadastrar - Validar cadastro com todos os campos preenchidos.	Os combobox estado deve estar carregado. O botão salvar estar habilitado.	01	Preencher todos os campos obrigatórios.	Clicar no botão <i>Cadastrar</i> .	O sistema grava as informações no sistema e exibe a mensagem "Registro salvo com Sucesso" e permanece na mesma tela. O teste case é finalizado.	
03	Cadastrar - Validar campos com máximo de caracteres preenchidos.	Os combobox estado deve estar carregado. O botão salvar estar habilitado.	01	Preencher todos os campos com máximo de caracteres.	Clicar no botão <i>Cadastrar</i> .	O sistema grava as informações no sistema e exibe a mensagem "Registro salvo com Sucesso" e permanece na mesma tela. O teste case é finalizado.	
04	Cadastrar - Validar preenchimento de campos com caracteres inválidos.	O botão salvar estar habilitado.	01	Preencher todos os campos com caracteres inválidos.	Clicar no botão <i>Cadastrar</i> .	O sistema não permite preencher os campos com caracteres inválidos. O teste case é finalizado.	
05	Editar – Validar alterações das informações.	Os combobox estado deve estar carregado. Editar um registro salvo.	01	Alterar todos os campos habilitados.	Pressionar o botão "Editar"	O sistema edita as informações no sistema e exibe a mensagem "Registro editado com Sucesso" e permanece na mesma tela. O teste case é finalizado.	

06	Editar – Validar campo em branco	Editar um registro salvo.	01	Limpar todos os campos habilitados.	Pressionar o botão “Editar”	O sistema não permite ação e exibe a mensagem “Não podem haver campos vazios”. O teste case é finalizado	
07	Editar - Validar campos com máximo de caracteres preenchidos.	O Os combobox estado deve estar carregado. O botão editar estar habilitado.	01	Preencher todos os campos com máximo de caracteres.	Clicar no botão “ <i>Editar</i> ”.	O sistema grava as informações e exibe a mensagem “Alteração realizada com sucesso” e permanece na mesma tela. O teste case é finalizado.	
08	Editar - Validar preenchimento de campos com caracteres inválidos.	O botão editar estar habilitado.	01	Preencher todos os campos com caracteres inválidos.	Clicar no botão “ <i>Editar</i> ”.	O sistema não permite preencher os campos com caracteres inválidos. O teste case é finalizado.	

Teste Case - UC008 – Manter Usuário

Casos de Teste

Use Case	UC008 – Cadastrar Usuário	Descrição
Número	Cenário	
01	Cadastrar - Salvar registro sem preencher nenhum campo.	

02	Cadastrar - Validar cadastro com todos os campos preenchidos.	
03	Cadastrar - Validar campos com máximo de caracteres preenchidos.	
04	Cadastrar - Validar preenchimento de campos com caracteres inválidos.	
Use Case	UC008 – Editar Usuário	Descrição
Número	Cenário	
05	Editar – Validar alterações das informações.	Salvar um registro e editar o mesmo.
06	Editar – Validar campo em branco	Limpar todos os campos.
07	Editar - Validar campos com máximo de caracteres preenchidos.	
08	Editar - Validar preenchimento de campos com caracteres inválidos.	

Caso de Uso		UC008 – Cadastrar Usuário					
Pré-condições		Ter acesso ao sistema. O sistema deverá ter executado o caso de uso Novo ou Alterar. O usuário deve possuir permissões para Salvar os dados no sistema.					
Elaborador		Maicon Alexander Illes Fang			Data de Elaboração		13/08/2013
Executor		Maicon Alexander Illes Fang			Data de Execução		15/05/2014
N.º	Nome	Pré-condições	Passo	Entrada	Ação	Resultado Esperado	Status
01	Cadastrar - Salvar registro sem preencher nenhum campo.	Os combobox perfil deve estar carregado. O botão salvar estar habilitado.	01	Não preencher nenhum campo.	Clicar no botão <i>Cadastrar</i> .	O sistema exibe a Mensagem “Não podem haver campos vazios”. O teste case é finalizado.	

02	Cadastrar - Validar cadastro com todos os campos preenchidos.	Os combobox perfil deve estar carregado. O botão salvar estar habilitado.	01	Preencher todos os campos obrigatórios.	Clicar no botão <i>Cadastrar</i> .	O sistema grava as informações no sistema e exibe a mensagem "Registro salvo com Sucesso" e permanece na mesma tela. O teste case é finalizado.	
03	Cadastrar - Validar campos com máximo de caracteres preenchidos.	Os combobox perfil deve estar carregado. O botão salvar estar habilitado.	01	Preencher todos os campos com máximo de caracteres.	Clicar no botão <i>Cadastrar</i> .	O sistema grava as informações no sistema e exibe a mensagem "Registro salvo com Sucesso" e permanece na mesma tela. O teste case é finalizado.	
04	Cadastrar - Validar preenchimento de campos com caracteres inválidos.	O botão salvar estar habilitado.	01	Preencher todos os campos com caracteres inválidos.	Clicar no botão <i>Cadastrar</i> .	O sistema não permite preencher os campos com caracteres inválidos. O teste case é finalizado.	
05	Editar – Validar alterações das informações.	Os combobox perfil deve estar carregado. Editar um registro salvo.	01	Alterar todos os campos habilitados.	Pressionar o botão "Editar"	O sistema edita as informações no sistema e exibe a mensagem "Registro editado com Sucesso" e permanece na mesma tela. O teste case é finalizado.	

06	Editar – Validar campo em branco	Editar um registro salvo.	01	Limpar todos os campos habilitados.	Pressionar o botão “Editar”	O sistema não permite ação e exibe a mensagem “Não podem haver campos vazios”. O teste case é finalizado	
07	Editar - Validar campos com máximo de caracteres preenchidos.	O Os combobox perfil deve estar carregado. O botão editar estar habilitado.	01	Preencher todos os campos com máximo de caracteres.	Clicar no botão “ <i>Editar</i> ”.	O sistema grava as informações e exibe a mensagem “Alteração realizada com sucesso” e permanece na mesma tela. O teste case é finalizado.	
08	Editar - Validar preenchimento de campos com caracteres inválidos.	O botão editar estar habilitado.	01	Preencher todos os campos com caracteres inválidos.	Clicar no botão “ <i>Editar</i> ”.	O sistema não permite preencher os campos com caracteres inválidos. O teste case é finalizado.	

Teste Case - UC009 – Manter Perfil

Casos de Teste

Use Case	UC009 – Cadastrar Perfil	Descrição
Número	Cenário	
01	Cadastrar - Salvar registro sem preencher nenhum campo.	

02	Cadastrar - Validar cadastro com todos os campos preenchidos.	
03	Cadastrar - Validar campos com máximo de caracteres preenchidos.	
04	Cadastrar - Validar preenchimento de campos com caracteres inválidos.	
Use Case	UC009 – Editar Perfil	Descrição
Número	Cenário	
05	Editar – Validar alterações das informações.	Salvar um registro e editar o mesmo.
06	Editar – Validar campo em branco	Limpar todos os campos.
07	Editar - Validar campos com máximo de caracteres preenchidos.	
08	Editar - Validar preenchimento de campos com caracteres inválidos.	

Caso de Uso			UC009 – Cadastrar Perfil				
Pré-condições			Ter acesso ao sistema. O sistema deverá ter executado o caso de uso Novo ou Alterar. O usuário deve possuir permissões para Salvar os dados no sistema.				
Elaborador			Maicon Alexander Illes Fang		Data de Elaboração		13/08/2013
Executor			Maicon Alexander Illes Fang		Data de Execução		15/05/2014
N.º	Nome	Pré-condições	Passo	Entrada	Ação	Resultado Esperado	Status
01	Cadastrar - Salvar registro sem preencher nenhum campo.	O botão salvar estar habilitado.	01	Não preencher nenhum campo.	Clicar no botão <i>Cadastrar</i> .	O sistema exibe a Mensagem "Não podem haver campos vazios". O teste case é finalizado.	

02	Cadastrar - Validar cadastro com todos os campos preenchidos.	O botão salvar estar habilitado.	01	Preencher todos os campos obrigatórios.	Clicar no botão <i>Cadastrar</i> .	O sistema grava as informações no sistema e exibe a mensagem "Registro salvo com Sucesso" e permanece na mesma tela. O teste case é finalizado.	
03	Cadastrar - Validar campos com máximo de caracteres preenchidos.	O botão salvar estar habilitado.	01	Preencher todos os campos com máximo de caracteres.	Clicar no botão <i>Cadastrar</i> .	O sistema grava as informações no sistema e exibe a mensagem "Registro salvo com Sucesso" e permanece na mesma tela. O teste case é finalizado.	
04	Cadastrar - Validar preenchimento de campos com caracteres inválidos.	O botão salvar estar habilitado.	01	Preencher todos os campos com caracteres inválidos.	Clicar no botão <i>Cadastrar</i> .	O sistema não permite preencher os campos com caracteres inválidos. O teste case é finalizado.	
05	Editar – Validar alterações das informações.	Editar um registro salvo.	01	Alterar todos os campos habilitados.	Pressionar o botão "Editar"	O sistema edita as informações no sistema e exibe a mensagem "Registro editado com Sucesso" e permanece na mesma tela. O teste case é finalizado.	

06	Editar – Validar campo em branco	Editar um registro salvo.	01	Limpar todos os campos habilitados.	Pressionar o botão “Editar”	O sistema não permite ação e exibe a mensagem “Não podem haver campos vazios”. O teste case é finalizado	
07	Editar - Validar campos com máximo de caracteres preenchidos.	O botão editar estar habilitado.	01	Preencher todos os campos com máximo de caracteres.	Clicar no botão “ <i>Editar</i> ”.	O sistema grava as informações e exibe a mensagem “Alteração realizada com sucesso” e permanece na mesma tela. O teste case é finalizado.	
08	Editar - Validar preenchimento de campos com caracteres inválidos.	O botão editar estar habilitado.	01	Preencher todos os campos com caracteres inválidos.	Clicar no botão “ <i>Editar</i> ”.	O sistema não permite preencher os campos com caracteres inválidos. O teste case é finalizado.	

Teste Case - UC010 – Manter Atividade

Casos de Teste

Use Case	UC010 – Cadastrar Atividade	Descrição
Número	Cenário	
01	Cadastrar - Salvar registro sem preencher nenhum campo.	

02	Cadastrar - Validar cadastro com todos os campos preenchidos.	
03	Cadastrar - Validar campos com máximo de caracteres preenchidos.	
04	Cadastrar - Validar preenchimento de campos com caracteres inválidos.	
Use Case	UC010 – Editar Atividade	Descrição
Número	Cenário	
05	Editar – Validar alterações das informações.	Salvar um registro e editar o mesmo.
06	Editar – Validar campo em branco	Limpar todos os campos.
07	Editar - Validar campos com máximo de caracteres preenchidos.	
08	Editar - Validar preenchimento de campos com caracteres inválidos.	

Caso de Uso		UC010 – Cadastrar Atividade					
Pré-condições		Ter acesso ao sistema. O sistema deverá ter executado o caso de uso Novo ou Alterar. O usuário deve possuir permissões para Salvar os dados no sistema.					
Elaborador		Maicon Alexander Illes Fang			Data de Elaboração		13/08/2013
Executor		Maicon Alexander Illes Fang			Data de Execução		15/05/2014
N.º	Nome	Pré-condições	Passo	Entrada	Ação	Resultado Esperado	Status
01	Cadastrar - Salvar registro sem preencher nenhum campo.	O botão salvar estar habilitado.	01	Não preencher nenhum campo.	Clicar no botão <i>Cadastrar</i> .	O sistema exibe a Mensagem "Não podem haver campos vazios". O teste case é finalizado.	

02	Cadastrar - Validar cadastro com todos os campos preenchidos.	O botão salvar estar habilitado.	01	Preencher todos os campos obrigatórios.	Clicar no botão <i>Cadastrar</i> .	O sistema grava as informações no sistema e exibe a mensagem "Registro salvo com Sucesso" e permanece na mesma tela. O teste case é finalizado.	
03	Cadastrar - Validar campos com máximo de caracteres preenchidos.	O botão salvar estar habilitado.	01	Preencher todos os campos com máximo de caracteres.	Clicar no botão <i>Cadastrar</i> .	O sistema grava as informações no sistema e exibe a mensagem "Registro salvo com Sucesso" e permanece na mesma tela. O teste case é finalizado.	
04	Cadastrar - Validar preenchimento de campos com caracteres inválidos.	O botão salvar estar habilitado.	01	Preencher todos os campos com caracteres inválidos.	Clicar no botão <i>Cadastrar</i> .	O sistema não permite preencher os campos com caracteres inválidos. O teste case é finalizado.	
05	Editar – Validar alterações das informações.	Editar um registro salvo.	01	Alterar todos os campos habilitados.	Pressionar o botão "Editar"	O sistema edita as informações no sistema e exibe a mensagem "Registro editado com Sucesso" e permanece na mesma tela. O teste case é finalizado.	

06	Editar – Validar campo em branco	Editar um registro salvo.	01	Limpar todos os campos habilitados.	Pressionar o botão “Editar”	O sistema não permite ação e exibe a mensagem “Não podem haver campos vazios”. O teste case é finalizado	
07	Editar - Validar campos com máximo de caracteres preenchidos.	O botão editar estar habilitado.	01	Preencher todos os campos com máximo de caracteres.	Clicar no botão “ <i>Editar</i> ”.	O sistema grava as informações e exibe a mensagem “Alteração realizada com sucesso” e permanece na mesma tela. O teste case é finalizado.	
08	Editar - Validar preenchimento de campos com caracteres inválidos.	O botão editar estar habilitado.	01	Preencher todos os campos com caracteres inválidos.	Clicar no botão “ <i>Editar</i> ”.	O sistema não permite preencher os campos com caracteres inválidos. O teste case é finalizado.	

Teste Case - UC011 – Manter Login

Casos de Teste

Use Case	UC011 – Login	Descrição
Número	Cenário	
01	Validar usuário desativado.	

02	Validar usuário ativo.	
03	Validar preenchimento de campos com caracteres inválidos.	

Caso de Uso			UC011 – Cadastrar Login				
Pré-condições			Estar na tela de login do sistema.				
Elaborador			Maicon Alexander Illes Fang		Data de Elaboração	13/08/2013	
Executor			Maicon Alexander Illes Fang		Data de Execução	15/05/2014	
N.º	Nome	Pré-condições	Passo	Entrada	Ação	Resultado Esperado	Status
01	Validar usuário desativado.	Um usuário inativo no sistema	01	Preencher os campos login e senha	Clicar no botão <i>login</i> .	O sistema exibe a Mensagem “Usuário inativo no sistema”. O teste case é finalizado.	
02	Validar usuário ativo.	Um usuário inativo no sistema	01	Preencher os campos login e senha	Clicar no botão <i>login</i> .	O sistema exibe a página home. O teste case é finalizado.	
03	Validar preenchimento de campos com caracteres inválidos.	N/A	01	Preencher os campos login e senha	Clicar no botão <i>login</i> .	O sistema não permite acesso a página inicial. O teste case é finalizado.	

Teste Case - UC500 – Exibir empresa/s na aplicação Android

Casos de Teste

Use Case	UC500 – Empresa Android	Descrição
Número	Cenário	
01	Validar acesso a aplicação sem acesso à internet.	
02	Validar acesso a aplicação com Webservice indisponível.	
03	Validar exibição das empresas sem nenhuma empresa cadastrada.	
04	Validar empresas desativadas no sistema.	

Caso de Uso		UC500 – Empresa Android					
Pré-condições		Estar com a aplicação instalada no dispositivo móvel.					
Elaborador		Maicon Alexander Illes Fang			Data de Elaboração		13/08/2013
Executor		Maicon Alexander Illes Fang			Data de Execução		15/05/2014
N.º	Nome	Pré-condições	Passo	Entrada	Ação	Resultado Esperado	Status
01	Validar acesso a aplicação sem acesso à internet.	Não ter conexão com internet.	01	Acessar a aplicação	N/A	A aplicação exibe a mensagem “Dispositivo sem conexão com a internet.” O teste case é finalizado.	
02	Validar acesso a aplicação com Webservice indisponível.	Não ter conexão com internet.	01	Acessar a aplicação	N/A	A aplicação exibe a mensagem “Webservice indisponível no momento.” O teste case é finalizado.	

03	Validar exibição das empresas sem nenhuma empresa cadastrada.	Não possuir empresa cadastrada no sistema web	01	Acessar a aplicação	N/A	A aplicação exibe a mensagem "Nenhuma empresa cadastrada." O teste case é finalizado.	
04	Validar empresas desativadas no sistema.	Não possuir empresa cadastrada no sistema web	01	Acessar a aplicação	N/A	A aplicação exibe a mensagem "Nenhuma empresa encontrada" O teste case é finalizado.	

Teste Case - UC501 – Exibir produto/s na aplicação Android

Casos de Teste

Use Case	UC501 – Produto Android	Descrição
Número	Cenário	
01	Validar acesso a aplicação sem acesso à internet.	
02	Validar acesso a aplicação com Webservice indisponível.	
03	Validar exibição do produto sem nenhum produto cadastrado.	
04	Validar produtos desativadas no sistema.	

Caso de Uso	UC501 – Produto Android		
Pré-condições	Estar com a aplicação instalada no dispositivo móvel.		
Elaborador	Maicon Alexander Illes Fang	Data de Elaboração	13/08/2013

Executor		Maicon Alexander Illes Fang			Data de Execução		15/05/2014
N. o	Nome	Pré- condições	Passo	Entrada	Ação	Resultado Esperado	Status
01	Validar acesso a aplicação sem acesso à internet.	Não ter conexão com internet.	01	Acessar a aplicação	N/A	A aplicação exibe a mensagem "Dispositivo sem conexão com a internet." O teste case é finalizado.	
02	Validar acesso a aplicação com WebService indisponível.	Não ter conexão com internet.	01	Acessar a aplicação	N/A	A aplicação exibe a mensagem "WebService indisponível no momento." O teste case é finalizado.	
03	Validar exibição do produto sem nenhum produto cadastrado.	Não possuir produto cadastrada no sistema web	01	Acessar a aplicação	N/A	A aplicação exibe a mensagem "Nenhum produto cadastrada." O teste case é finalizado.	
04	Validar produtos desativadas no sistema.	Não possuir produto cadastrada no sistema web	01	Acessar a aplicação	N/A	A aplicação exibe a mensagem "Nenhum produto encontrada" O teste case é finalizado.	

Teste Case - UC502 – Exibir detalhe de um produto na aplicação Android

Casos de Teste

Use Case	UC502 – Detalhe do Produto Android	Descrição
Número	Cenário	

01	Validar layout da tela.	
----	-------------------------	--

Caso de Uso		UC500 – Empresa Android					
Pré-condições		Estar com a aplicação instalada no dispositivo móvel.					
Elaborador		Maicon Alexander Illes Fang			Data de Elaboração	13/08/2013	
Executor		Maicon Alexander Illes Fang			Data de Execução	15/05/2014	
N.º	Nome	Pré-condições	Passo	Entrada	Ação	Resultado Esperado	Status
01	Validar layout da tela.	Ter uma empresa ativa. Ter um produto ativo.	01	Acessar a aplicação	N/A	A aplicação exibe os dados corretamente na tela. O teste case é finalizado.	

APÊNDICE Q 17 - Build – Classes, programas, etc. Software programado.

Build – Classes, programas, etc. Software programado

Versão 2.1

Histórico da Revisão

Data	Versão	Descrição	Autor
------	--------	-----------	-------

10/09/2014	1.0	Versão inicial	Maicon Alexander Illes Fang
------------	-----	----------------	--------------------------------

Especificação de Caso de Teste

Teste Case - UC001 – Manter Empresa

```
package com.mBestProducts.bean;
import java.util.Date;
public class Empresa {
    private int    empEmpresalD;
    private String empEmpresa;
    private String empCNPJ;
    private String empIE;
    private String empEndCEP;
    private String empEndLogradouro;
    private String empEndBairro;
    private String empEndNumero;
    private String empEndComplemento;
    private String empConContato;
    private String empConEmail;
```

```
private String empConRedeSocial;
private String empConSite;
private String empLocalizacao;
private String empLogo; // endereço repositório de imagem
private Date empDataCadastro;
private Date empUltimaAlteracao;
private boolean empAtivo;
private String empRanking; // Envie como String, o banco cuidará de armazenar como char.

public Cidade cidade;
public Atividade atividade;

public Empresa (){
    empDataCadastro = new Date();
    empUltimaAlteracao = new Date();
    atividade = new Atividade();
    cidade = new Cidade();
}

// Usado somente no método que retorna dados para android
public Empresa (String teste){
    String aux = teste;
```

```
}
```

```
/**
```

```
 * @return the empEmpresalD
```

```
 */
```

```
public int getEmpEmpresalD() {
```

```
    return empEmpresalD;
```

```
}
```

```
/**
```

```
 * @param empEmpresalD the empEmpresalD to set
```

```
 */
```

```
public void setEmpEmpresalD(int empEmpresalD) {
```

```
    this.empEmpresalD = empEmpresalD;
```

```
}
```

```
/**
```

```
 * @return the empEmpresa
```

```
 */
```

```
public String getEmpEmpresa() {
```

```
    return empEmpresa;
```

```
}
```



```
/**  
 * @param empEmpresa the empEmpresa to set  
 */  
public void setEmpEmpresa(String empEmpresa) {  
    this.empEmpresa = empEmpresa;  
}
```

```
/**  
 * @return the empCNPJ  
 */  
public String getEmpCNPJ() {  
    return empCNPJ;  
}
```

```
/**  
 * @param empCNPJ the empCNPJ to set  
 */  
public void setEmpCNPJ(String empCNPJ) {  
    this.empCNPJ = empCNPJ;  
}
```

```
/**
 * @return the emplE
 */
public String getEmplE() {
    return emplE;
}

/**
 * @param emplE the emplE to set
 */
public void setEmplE(String emplE) {
    this.emplE = emplE;
}

/**
 * @return the empEndCEP
 */
public String getEmpEndCEP() {
    return empEndCEP;
}

/**
```

```
* @param empEndCEP the empEndCEP to set
*/
public void setEmpEndCEP(String empEndCEP) {
    this.empEndCEP = empEndCEP;
}

/**
 * @return the empEndLogradouro
 */
public String getEmpEndLogradouro() {
    return empEndLogradouro;
}

/**
 * @param empEndLogradouro the empEndLogradouro to set
 */
public void setEmpEndLogradouro(String empEndLogradouro) {
    this.empEndLogradouro = empEndLogradouro;
}

/**
 * @return the empEndBairro
```

```
*/  
public String getEmpEndBairro() {  
    return empEndBairro;  
}  
  
/**  
 * @param empEndBairro the empEndBairro to set  
 */  
public void setEmpEndBairro(String empEndBairro) {  
    this.empEndBairro = empEndBairro;  
}  
  
/**  
 * @return the empEndNumero  
 */  
public String getEmpEndNumero() {  
    return empEndNumero;  
}  
  
/**  
 * @param empEndNumero the empEndNumero to set  
 */
```

```
public void setEmpEndNumero(String empEndNumero) {
    this.empEndNumero = empEndNumero;
}

/**
 * @return the empEndComplemento
 */
public String getEmpEndComplemento() {
    return empEndComplemento;
}

/**
 * @param empEndComplemento the empEndComplemento to set
 */
public void setEmpEndComplemento(String empEndComplemento) {
    this.empEndComplemento = empEndComplemento;
}

/**
 * @return the empConContato
 */
public String getEmpConContato() {
```

```
        return empConContato;
    }

    /**
     * @param empConContato the empConContato to set
     */
    public void setEmpConContato(String empConContato) {
        this.empConContato = empConContato;
    }

    /**
     * @return the empConEmail
     */
    public String getEmpConEmail() {
        return empConEmail;
    }

    /**
     * @param empConEmail the empConEmail to set
     */
    public void setEmpConEmail(String empConEmail) {
        this.empConEmail = empConEmail;
    }
}
```

```
}

/**
 * @return the empConRedeSocial
 */
public String getEmpConRedeSocial() {
    return empConRedeSocial;
}

/**
 * @param empConRedeSocial the empConRedeSocial to set
 */
public void setEmpConRedeSocial(String empConRedeSocial) {
    this.empConRedeSocial = empConRedeSocial;
}

/**
 * @return the empConSite
 */
public String getEmpConSite() {
    return empConSite;
}
```

```
/**
 * @param empConSite the empConSite to set
 */
public void setEmpConSite(String empConSite) {
    this.empConSite = empConSite;
}

/**
 * @return the empLocalizacao
 */
public String getEmpLocalizacao() {
    return empLocalizacao;
}

/**
 * @param empLocalizacao the empLocalizacao to set
 */
public void setEmpLocalizacao(String empLocalizacao) {
    this.empLocalizacao = empLocalizacao;
}
```



```
/**
 * @return the empLogo
 */
public String getEmpLogo() {
    return empLogo;
}

/**
 * @param empLogo the empLogo to set
 */
public void setEmpLogo(String empLogo) {
    this.empLogo = empLogo;
}

/**
 * @return the empDataCadastro
 */
public Date getEmpDataCadastro() {
    return empDataCadastro;
}

/**
```

```
* @param empDataCadastro the empDataCadastro to set
*/
public void setEmpDataCadastro(Date empDataCadastro) {
    this.empDataCadastro = empDataCadastro;
}

/**
 * @return the empUltimaAlteracao
 */
public Date getEmpUltimaAlteracao() {
    return empUltimaAlteracao;
}

/**
 * @param empUltimaAlteracao the empUltimaAlteracao to set
 */
public void setEmpUltimaAlteracao(Date empUltimaAlteracao) {
    this.empUltimaAlteracao = empUltimaAlteracao;
}

/**
 * @return the empAtivo
```

```
*/  
public boolean isEmpAtivo() {  
    return empAtivo;  
}  
  
/**  
 * @param empAtivo the empAtivo to set  
 */  
public void setEmpAtivo(boolean empAtivo) {  
    this.empAtivo = empAtivo;  
}  
  
/**  
 * @return the empRanking  
 */  
public String getEmpRanking() {  
    return empRanking;  
}  
  
/**  
 * @param empRanking the empRanking to set  
 */
```

```
public void setEmpRanking(String empRanking) {  
    this.empRanking = empRanking;  
}
```

```
/**
```

```
 * @return the cidade
```

```
 */
```

```
public Cidade getCidade() {  
    return cidade;  
}
```

```
/**
```

```
 * @param cidade the cidade to set
```

```
 */
```

```
public void setCidade(Cidade cidade) {  
    this.cidade = cidade;  
}
```

```
/**
```

```
 * @return the atividade
```

```
 */
```

```
public Atividade getAtividade() {
```

```
        return atividade;
    }

    /**
     * @param atividade the atividade to set
     */
    public void setAtividade(Atividade atividade) {
        this.atividade = atividade;
    }
}
```

Teste Case - UC002 – Manter Funcionário

```
package com.mBestProducts.bean;
import java.util.Date;
public class Funcionario {
    private int    funFuncionarioID;
    private String funFuncionario;
    private String funCPF;
    private String funRG;
    private Date   funDataNascimento;
    private String funEndCEP;
```

```
private String funEndLogradouro;
private String funEndBairro;
private String funEndNumero;
private String funEndComplemento;
private String funConContato;
private String funConEmail;
private String funConRedeSocial;
private boolean funAtivo;
private Date funDataCadastro;
private Date funUltimaAlteracao;

public Cidade cidade;
public Empresa empresa;

public Funcionario (){
    funDataCadastro = new Date();
    funUltimaAlteracao = new Date();
    empresa = new Empresa();
    cidade = new Cidade();
}

/**
```

```
* @return the funFuncionarioID
*/
public int getFunFuncionarioID() {
    return funFuncionarioID;
}

/**
 * @param funFuncionarioID the funFuncionarioID to set
 */
public void setFunFuncionarioID(int funFuncionarioID) {
    this.funFuncionarioID = funFuncionarioID;
}

/**
 * @return the funFuncionario
 */
public String getFunFuncionario() {
    return funFuncionario;
}

/**
 * @param funFuncionario the funFuncionario to set
```

```
*/  
public void setFunFuncionario(String funFuncionario) {  
    this.funFuncionario = funFuncionario;  
}
```

```
/**  
 * @return the funCPF  
 */  
public String getFunCPF() {  
    return funCPF;  
}
```

```
/**  
 * @param funCPF the funCPF to set  
 */  
public void setFunCPF(String funCPF) {  
    this.funCPF = funCPF;  
}
```

```
/**  
 * @return the funRG  
 */
```



```
public String getFunRG() {  
    return funRG;  
}
```

```
/**
```

```
 * @param funRG the funRG to set
```

```
 */
```

```
public void setFunRG(String funRG) {  
    this.funRG = funRG;  
}
```

```
/**
```

```
 * @return the funDataNascimento
```

```
 */
```

```
public Date getFunDataNascimento() {  
    return funDataNascimento;  
}
```

```
/**
```

```
 * @param funDataNascimento the funDataNascimento to set
```

```
 */
```

```
public void setFunDataNascimento(Date funDataNascimento) {
```

```
        this.funDataNascimento = funDataNascimento;
    }

    /**
     * @return the funEndCEP
     */
    public String getFunEndCEP() {
        return funEndCEP;
    }

    /**
     * @param funEndCEP the funEndCEP to set
     */
    public void setFunEndCEP(String funEndCEP) {
        this.funEndCEP = funEndCEP;
    }

    /**
     * @return the funEndLogradouro
     */
    public String getFunEndLogradouro() {
        return funEndLogradouro;
    }
}
```

```
}

/**
 * @param funEndLogradouro the funEndLogradouro to set
 */
public void setFunEndLogradouro(String funEndLogradouro) {
    this.funEndLogradouro = funEndLogradouro;
}

/**
 * @return the funEndBairro
 */
public String getFunEndBairro() {
    return funEndBairro;
}

/**
 * @param funEndBairro the funEndBairro to set
 */
public void setFunEndBairro(String funEndBairro) {
    this.funEndBairro = funEndBairro;
}
```

```
/**
 * @return the funEndNumero
 */
public String getFunEndNumero() {
    return funEndNumero;
}

/**
 * @param funEndNumero the funEndNumero to set
 */
public void setFunEndNumero(String funEndNumero) {
    this.funEndNumero = funEndNumero;
}

/**
 * @return the funEndComplemento
 */
public String getFunEndComplemento() {
    return funEndComplemento;
}
```

```
/**
 * @param funEndComplemento the funEndComplemento to set
 */
public void setFunEndComplemento(String funEndComplemento) {
    this.funEndComplemento = funEndComplemento;
}
```

```
/**
 * @return the funConContato
 */
public String getFunConContato() {
    return funConContato;
}
```

```
/**
 * @param funConContato the funConContato to set
 */
public void setFunConContato(String funConContato) {
    this.funConContato = funConContato;
}
```

```
/**
```

```
* @return the funConEmail
*/
public String getFunConEmail() {
    return funConEmail;
}

/**
 * @param funConEmail the funConEmail to set
 */
public void setFunConEmail(String funConEmail) {
    this.funConEmail = funConEmail;
}

/**
 * @return the funConRedeSocial
 */
public String getFunConRedeSocial() {
    return funConRedeSocial;
}

/**
 * @param funConRedeSocial the funConRedeSocial to set
```

```
*/  
public void setFunConRedeSocial(String funConRedeSocial) {  
    this.funConRedeSocial = funConRedeSocial;  
}
```

```
/**  
 * @return the funAtivo  
 */  
public boolean isFunAtivo() {  
    return funAtivo;  
}
```

```
/**  
 * @param funAtivo the funAtivo to set  
 */  
public void setFunAtivo(boolean funAtivo) {  
    this.funAtivo = funAtivo;  
}
```

```
/**  
 * @return the funDataCadastro  
 */
```

```
public Date getFunDataCadastro() {  
    return funDataCadastro;  
}  
  
/**  
 * @param funDataCadastro the funDataCadastro to set  
 */  
public void setFunDataCadastro(Date funDataCadastro) {  
    this.funDataCadastro = funDataCadastro;  
}  
  
/**  
 * @return the funUltimaAlteracao  
 */  
public Date getFunUltimaAlteracao() {  
    return funUltimaAlteracao;  
}  
  
/**  
 * @param funUltimaAlteracao the funUltimaAlteracao to set  
 */  
public void setFunUltimaAlteracao(Date funUltimaAlteracao) {
```



```
        this.funUltimaAlteracao = funUltimaAlteracao;
    }

    /**
     * @return the cidade
     */
    public Cidade getCidade() {
        return cidade;
    }

    /**
     * @param cidade the cidade to set
     */
    public void setCidade(Cidade cidade) {
        this.cidade = cidade;
    }

    /**
     * @return the empresa
     */
    public Empresa getEmpresa() {
        return empresa;
    }
}
```

```
}

/**
 * @param empresa the empresa to set
 */
public void setEmpresa(Empresa empresa) {
    this.empresa = empresa;
}
}
```

Teste Case - UC003 – Manter Produto

```
package com.mBestProducts.bean;
import java.util.Date;
public class Produto {
    private int proProdutoID;
    private String proProduto;
    private String proDescricao;
    private String proInfTecnicas;
    private int proQuantidade;
    private float proPrecoNormal;
    private float proPrecoPromocao;
```

```
private Date proDataValidade;  
private String proRanking;  
private boolean proAtivo;  
private Date proDataCadastro;  
private Date proUltimaAlteracao;  
private String proFoto;  
private byte[] foto;
```

```
private String strDataValidade;
```

```
public Funcionario funcionario;  
public Empresa empresa;  
public Categoria categoria;
```

```
public Produto (){  
    proDataCadastro = new Date();  
    proUltimaAlteracao = new Date();  
    funcionario = new Funcionario();  
    empresa = new Empresa();
```

```
        categoria = new Categoria();  
    }
```

// Usado somente no metodo que retorna dados para android

```
public Produto (String teste){  
    String aux = teste;  
}
```

```
public Produto (int id){  
    proProdutoID = id;  
    empresa = new Empresa();  
    categoria = new Categoria();  
}
```

```
/**
```

```
 * @return the strDataValidade
```

```
 */
```

```
public String getStrDataValidade() {  
    return strDataValidade;  
}
```

```
/**
 * @param strDataValidade the strDataValidade to set
 */
public void setStrDataValidade(String strDataValidade) {
    this.strDataValidade = strDataValidade;
}
```

```
/**
 * @return the categoria
 */
public Categoria getCategoria() {
    return categoria;
}
```

```
/**
 * @param categoria the categoria to set
 */
public void setCategoria(Categoria categoria) {
    this.categoria = categoria;
}
```

```
/**
 * @return the proProdutoID
 */
public int getProProdutoID() {
    return proProdutoID;
}

/**
 * @param proProdutoID the proProdutoID to set
 */
public void setProProdutoID(int proProdutoID) {
    this.proProdutoID = proProdutoID;
}

/**
 * @return the proProduto
 */
public String getProProduto() {
    return proProduto;
}

/**
```

```
* @param proProduto the proProduto to set
*/
public void setProProduto(String proProduto) {
    this.proProduto = proProduto;
}

/**
 * @return the proDescricao
 */
public String getProDescricao() {
    return proDescricao;
}

/**
 * @param proDescricao the proDescricao to set
 */
public void setProDescricao(String proDescricao) {
    this.proDescricao = proDescricao;
}

/**
 * @return the proInfTecnicas
```

```
*/  
public String getProInfTecnicas() {  
    return proInfTecnicas;  
}  
  
/**  
 * @param proInfTecnicas the proInfTecnicas to set  
 */  
public void setProInfTecnicas(String proInfTecnicas) {  
    this.proInfTecnicas = proInfTecnicas;  
}  
  
/**  
 * @return the proQuantidade  
 */  
public int getProQuantidade() {  
    return proQuantidade;  
}  
  
/**  
 * @param proQuantidade the proQuantidade to set  
 */
```



```
public void setProQuantidade(int proQuantidade) {
    this.proQuantidade = proQuantidade;
}

/**
 * @return the proPrecoNormal
 */
public float getProPrecoNormal() {
    return proPrecoNormal;
}

/**
 * @param proPrecoNormal the proPrecoNormal to set
 */
public void setProPrecoNormal(float proPrecoNormal) {
    this.proPrecoNormal = proPrecoNormal;
}

/**
 * @return the proPrecoPromocao
 */
public float getProPrecoPromocao() {
```

```
        return proPrecoPromocao;
    }

    /**
     * @param proPrecoPromocao the proPrecoPromocao to set
     */
    public void setProPrecoPromocao(float proPrecoPromocao) {
        this.proPrecoPromocao = proPrecoPromocao;
    }

    /**
     * @return the proDataValidade
     */
    public Date getProDataValidade() {
        return proDataValidade;
    }

    /**
     * @param proDataValidade the proDataValidade to set
     */
    public void setProDataValidade(Date proDataValidade) {
        this.proDataValidade = proDataValidade;
    }
}
```

```
}
```

```
/**
```

```
 * @return the proRanking
```

```
 */
```

```
public String getProRanking() {
```

```
    return proRanking;
```

```
}
```

```
/**
```

```
 * @param proRanking the proRanking to set
```

```
 */
```

```
public void setProRanking(String proRanking) {
```

```
    this.proRanking = proRanking;
```

```
}
```

```
/**
```

```
 * @return the proAtivo
```

```
 */
```

```
public boolean isProAtivo() {
```

```
    return proAtivo;
```

```
}
```

```
/**
 * @param proAtivo the proAtivo to set
 */
public void setProAtivo(boolean proAtivo) {
    this.proAtivo = proAtivo;
}

/**
 * @return the proDataCadastro
 */
public Date getProDataCadastro() {
    return proDataCadastro;
}

/**
 * @param proDataCadastro the proDataCadastro to set
 */
public void setProDataCadastro(Date proDataCadastro) {
    this.proDataCadastro = proDataCadastro;
}
```

```
/**
 * @return the proUltimaAlteracao
 */
public Date getProUltimaAlteracao() {
    return proUltimaAlteracao;
}

/**
 * @param proUltimaAlteracao the proUltimaAlteracao to set
 */
public void setProUltimaAlteracao(Date proUltimaAlteracao) {
    this.proUltimaAlteracao = proUltimaAlteracao;
}

/**
 * @return the funcionario
 */
public Funcionario getFuncionario() {
    return funcionario;
}

/**
```

```
* @param funcionario the funcionario to set
*/
public void setFuncionario(Funcionario funcionario) {
    this.funcionario = funcionario;
}
```

```
/**
 * @return the empresa
 */
public Empresa getEmpresa() {
    return empresa;
}
```

```
/**
 * @param empresa the empresa to set
 */
public void setEmpresa(Empresa empresa) {
    this.empresa = empresa;
}
```

```
public String getProFoto() {
    return proFoto;
}
```

```
}
```

```
public void setProFoto(String proFoto) {  
    this.proFoto = proFoto;  
}
```

```
/**
```

```
 * @return the foto
```

```
 */
```

```
public byte[] getFoto() {  
    return foto;  
}
```

```
/**
```

```
 * @param foto the foto to set
```

```
 */
```

```
public void setFoto(byte[] foto) {  
    this.foto = foto;  
}  
}
```

Teste Case - UC004 – Manter Categoria

```
package com.mBestProducts.bean;
import java.util.Date;
public class Categoria {
    private int catCategorialD;
    private String catCategoria;
    private Date catDataCadastro;
    private Date catUltimaAlteracao;

    public Categoria (){
        catDataCadastro = new Date();
        catUltimaAlteracao = new Date();
    }

    /*
     * Getters e Setters
     */

    public int getCatCategorialD() {
        return catCategorialD;
    }
}
```



```
}  
public void setCatCategorialID(int catCategorialID) {  
    this.catCategorialID = catCategorialID;  
}  
public String getCatCategoria() {  
    return catCategoria;  
}  
public void setCatCategoria(String catCategoria) {  
    this.catCategoria = catCategoria;  
}  
public Date getCatDataCadastro() {  
    return catDataCadastro;  
}  
public void setCatDataCadastro(Date catDataCadastro) {  
    this.catDataCadastro = catDataCadastro;  
}  
public Date getCatUltimaAlteracao() {  
    return catUltimaAlteracao;  
}  
public void setCatUltimaAlteracao(Date catUltimaAlteracao) {  
    this.catUltimaAlteracao = catUltimaAlteracao;  
}
```

```
}
```

Teste Case - UC005 – Manter Pais

```
package com.mBestProducts.bean;
import java.util.Date;
public class Pais {
    private int paiPaisID;
    private String paiPais;
    private String paiDDI;
    private Date paiDataCadastro;
    private Date paiUltimaAlteracao;

    public Pais (){
        paiDataCadastro = new Date();
        paiUltimaAlteracao = new Date();
    }

    /**
     * @return the paiDDI
```

```
*/  
public String getPaiDDI() {  
    return paiDDI;  
}  
/**  
 * @param paiDDI the paiDDI to set  
 */  
public void setPaiDDI(String paiDDI) {  
    this.paiDDI = paiDDI;  
}  
/**  
 * @return the paiPaisID  
 */  
public int getPaiPaisID() {  
    return paiPaisID;  
}  
/**  
 * @param paiPaisID the paiPaisID to set  
 */  
public void setPaiPaisID(int paiPaisID) {  
    this.paiPaisID = paiPaisID;  
}
```

```
/**
 * @return the paiPais
 */
public String getPaiPais() {
    return paiPais;
}

/**
 * @param paiPais the paiPais to set
 */
public void setPaiPais(String paiPais) {
    this.paiPais = paiPais;
}

/**
 * @return the paiDataCadastro
 */
public Date getPaiDataCadastro() {
    return paiDataCadastro;
}

/**
 * @param paiDataCadastro the paiDataCadastro to set
 */
public void setPaiDataCadastro(Date paiDataCadastro) {
```

```

        this.paiDataCadastro = paiDataCadastro;
    }
    /**
     * @return the paiUltimaAlteracao
     */
    public Date getPaiUltimaAlteracao() {
        return paiUltimaAlteracao;
    }
    /**
     * @param paiUltimaAlteracao the paiUltimaAlteracao to set
     */
    public void setPaiUltimaAlteracao(Date paiUltimaAlteracao) {
        this.paiUltimaAlteracao = paiUltimaAlteracao;
    }
}

```

Teste Case - UC006 – Manter Estado

package com.mBestProducts.bean;

import java.util.Date;

public class Estado {

```
private int estEstadoID;
private String estEstado;
private String estUF;
private Date estDataCadastro;
private Date estUltimaAlteracao;
public Pais pais;

public Estado (){
    estDataCadastro = new Date();
    estUltimaAlteracao = new Date();
    pais = new Pais();
}

/**
 * @return the estEstadoID
 */
public int getEstEstadoID() {
    return estEstadoID;
}

/**
 * @param estEstadoID the estEstadoID to set
```

```
*/  
public void setEstEstadoID(int estEstadoID) {  
    this.estEstadoID = estEstadoID;  
}
```

```
/**  
 * @return the estEstado  
 */  
public String getEstEstado() {  
    return estEstado;  
}
```

```
/**  
 * @param estEstado the estEstado to set  
 */  
public void setEstEstado(String estEstado) {  
    this.estEstado = estEstado;  
}
```

```
/**  
 * @return the estUF  
 */
```

```
public String getEstUF() {  
    return estUF;  
}  
  
/**  
 * @param estUF the estUF to set  
 */  
public void setEstUF(String estUF) {  
    this.estUF = estUF;  
}  
  
/**  
 * @return the estDataCadaastro  
 */  
public Date getEstDataCadaastro() {  
    return estDataCadaastro;  
}  
  
/**  
 * @param estDataCadaastro the estDataCadaastro to set  
 */  
public void setEstDataCadaastro(Date estDataCadaastro) {
```



```
        this.estDataCadastro = estDataCadastro;
    }

    /**
     * @return the estUltimaAlteracao
     */
    public Date getEstUltimaAlteracao() {
        return estUltimaAlteracao;
    }

    /**
     * @param estUltimaAlteracao the estUltimaAlteracao to set
     */
    public void setEstUltimaAlteracao(Date estUltimaAlteracao) {
        this.estUltimaAlteracao = estUltimaAlteracao;
    }

    /**
     * @return the pais
     */
    public Pais getPais() {
        return pais;
    }
}
```

```
}

/**
 * @param pais the pais to set
 */
public void setPais(Pais pais) {
    this.pais = pais;
}
}
```

Teste Case - UC007 – Manter Cidade
package com.mBestProducts.bean;

```
import java.util.Date;
public class Cidade {
    private int cidCidadeID;
    private String cidCidade;
    private String cidDDD;
    private Date cidDataCadastro;
    private Date cidUltimaAlteracao;
    public Estado estado;
```

```
public Cidade (){
    cidDataCadastro = new Date();
    cidUltimaAlteracao = new Date();
    estado = new Estado();
}

/**
 * @return the cidCidadeID
 */
public int getCidCidadeID() {
    return cidCidadeID;
}

/**
 * @param cidCidadeID the cidCidadeID to set
 */
public void setCidCidadeID(int cidCidadeID) {
    this.cidCidadeID = cidCidadeID;
}

/**
```

```
* @return the cidCidade
*/
public String getCidCidade() {
    return cidCidade;
}

/**
 * @param cidCidade the cidCidade to set
 */
public void setCidCidade(String cidCidade) {
    this.cidCidade = cidCidade;
}

/**
 * @return the cidDDD
 */
public String getCidDDD() {
    return cidDDD;
}

/**
 * @param cidDDD the cidDDD to set
```

```
*/  
public void setCidDDD(String cidDDD) {  
    this.cidDDD = cidDDD;  
}  
  
/**  
 * @return the cidDataCadaastro  
 */  
public Date getCidDataCadaastro() {  
    return cidDataCadaastro;  
}  
  
/**  
 * @param cidDataCadaastro the cidDataCadaastro to set  
 */  
public void setCidDataCadaastro(Date cidDataCadaastro) {  
    this.cidDataCadaastro = cidDataCadaastro;  
}  
  
/**  
 * @return the cidUltimaAlteracao  
 */
```

```
public Date getCidUltimaAlteracao() {  
    return cidUltimaAlteracao;  
}  
  
/**  
 * @param cidUltimaAlteracao the cidUltimaAlteracao to set  
 */  
public void setCidUltimaAlteracao(Date cidUltimaAlteracao) {  
    this.cidUltimaAlteracao = cidUltimaAlteracao;  
}  
  
/**  
 * @return the estado  
 */  
public Estado getEstado() {  
    return estado;  
}  
  
/**  
 * @param estado the estado to set  
 */  
public void setEstado(Estado estado) {
```

```
        this.estado = estado;  
    }  
}
```

Teste Case - UC008 – Manter Usuário

```
package com.mBestProducts.bean;  
import java.util.Date;  
import javax.faces.context.FacesContext;  
public class Usuario {  
    private int usuID;  
    private String usuLogin;  
    private int usuMatricula;  
    private int id_perfil;  
    private String usuEmail;  
    private String usuSenha;  
    private String usuConfSenha;  
    private String usuDicaSenha;  
    private int usuTentativaLogin;  
    private String usuNome;  
    private boolean usuAtivo;
```

```
private Date usuDataCadastro;  
private Date usuUltimaAlteracao;  
private Perfil perfil;
```

```
//Getters e setters
```

```
/*((HttpServletRequest)  
FacesContext.getCurrentInstance().getExternalContext().getRequest()).getUserPrincipal().getName(); ((HttpServletRequest)  
FacesContext.getCurrentInstance().getExternalContext().getRequest()).getUserPrincipal().getName();  
    * O metodo criado abaixo testa se o objeto possui ou não o login.  
    */  
public boolean isPersisted() {  
    return this.getUsuLogin() != null;  
}  
  
public int getUsulD() {  
    return usulD;  
}  
  
public void setUsulD(int usulD) {  
    this.usulD = usulD;  
}
```



```
public String getUsuLogin() {  
    return usuLogin;  
}
```

```
public void setUsuLogin(String usuLogin) {  
    this.usuLogin = usuLogin;  
}
```

```
public int getUsuMatricula() {  
    return usuMatricula;  
}
```

```
public void setUsuMatricula(int usuMatricula) {  
    this.usuMatricula = usuMatricula;  
}
```

```
public int getId_perfil() {  
    return id_perfil;  
}
```

```
public void setId_perfil(int id_perfil) {
```

```
        this.id_perfil = id_perfil;
    }

    public String getUsuEmail() {
        return usuEmail;
    }

    public void setUsuEmail(String usuEmail) {
        this.usuEmail = usuEmail;
    }

    public String getUsuSenha() {
        return usuSenha;
    }

    public void setUsuSenha(String usuSenha) {
        this.usuSenha = usuSenha;
    }

    public String getUsuConfSenha() {
        return usuConfSenha;
    }
}
```

```
public void setUsuConfSenha(String usuConfSenha) {  
    this.usuConfSenha = usuConfSenha;  
}
```

```
public String getUsuDicaSenha() {  
    return usuDicaSenha;  
}
```

```
public void setUsuDicaSenha(String usuDicaSenha) {  
    this.usuDicaSenha = usuDicaSenha;  
}
```

```
public int getUsuTentativaLogin() {  
    return usuTentativaLogin;  
}
```

```
public void setUsuTentativaLogin(int usuTentativaLogin) {  
    this.usuTentativaLogin = usuTentativaLogin;  
}
```

```
public String getUsuNome() {  
    return usuNome;  
}
```

```
}
```

```
public void setUsuNome(String usuNome) {  
    this.usuNome = usuNome;  
}
```

```
public boolean isUsuAtivo() {  
    return usuAtivo;  
}
```

```
public void setUsuAtivo(boolean usuAtivo) {  
    this.usuAtivo = usuAtivo;  
}
```

```
public Date getUsuDataCadastro() {  
    return usuDataCadastro;  
}
```

```
public void setUsuDataCadastro(Date usuDataCadastro) {  
    this.usuDataCadastro = usuDataCadastro;  
}
```

```
public Date getUsuUltimaAlteracao() {  
    return usuUltimaAlteracao;  
}  
  
public void setUsuUltimaAlteracao(Date usuUltimaAlteracao) {  
    this.usuUltimaAlteracao = usuUltimaAlteracao;  
}  
  
public Perfil getPerfil() {  
    return perfil;  
}  
  
public void setPerfil(Perfil perfil) {  
    this.perfil = perfil;  
}  
}
```

Teste Case - UC009 – Manter Perfil
package com.mBestProducts.bean;

```
import java.util.Date;

public class Perfil {
    private int perPerfillID;
    private String perPerfil;
    private Date perDataCadastro;
    private Date perUltimaAlteracao;

    public Perfil (){
        perDataCadastro = new Date();
        perUltimaAlteracao = new Date();
    }

    /*
     * Getters e Setters
     */

    public int getPerPerfillID() {
        return perPerfillID;
    }

    public void setPerPerfillID(int perPerfillID) {
        this.perPerfillID = perPerfillID;
    }
}
```

```
public String getPerPerfil() {  
    return perPerfil;  
}  
public void setPerPerfil(String perPerfil) {  
    this.perPerfil = perPerfil;  
}  
public Date getPerDataCadastro() {  
    return perDataCadastro;  
}  
public void setPerDataCadastro(Date perDataCadastro) {  
    this.perDataCadastro = perDataCadastro;  
}  
public Date getPerUltimaAlteracao() {  
    return perUltimaAlteracao;  
}  
public void setPerUltimaAlteracao(Date perUltimaAlteracao) {  
    this.perUltimaAlteracao = perUltimaAlteracao;  
}  
}
```

```
package com.mBestProducts.bean;
import java.util.Date;
public class Atividade {
    private int atiAtividadeID;
    private String atiAtividade;
    private Date atiDataCadastro;
    private Date atiUltimaAlteracao;

    public Atividade (){
        atiDataCadastro = new Date();
        atiUltimaAlteracao = new Date();
    }
    /*
     * Getters e Setters
     */

    public int getAtiAtividadeID() {
        return atiAtividadeID;
    }
    public void setAtiAtividadeID(int atiAtividadeID) {
        this.atiAtividadeID = atiAtividadeID;
    }
}
```



```
}  
public String getAtiAtividade() {  
    return atiAtividade;  
}  
public void setAtiAtividade(String atiAtividade) {  
    this.atiAtividade = atiAtividade;  
}  
public Date getAtiDataCadastro() {  
    return atiDataCadastro;  
}  
public void setAtiDataCadastro(Date atiDataCadastro) {  
    this.atiDataCadastro = atiDataCadastro;  
}  
public Date getAtiUltimaAlteracao() {  
    return atiUltimaAlteracao;  
}  
public void setAtiUltimaAlteracao(Date atiUltimaAlteracao) {  
    this.atiUltimaAlteracao = atiUltimaAlteracao;  
}  
  
}
```

APÊNDICE R - 18 - Log de Testes – Resultado dos testes realizados.

18 - Log de Testes – Resultado dos testes realizados.

Versão 2.1

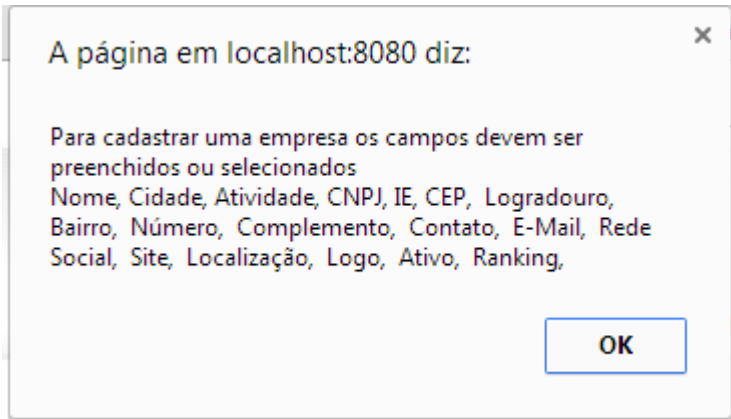


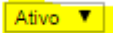
Histórico da Revisão


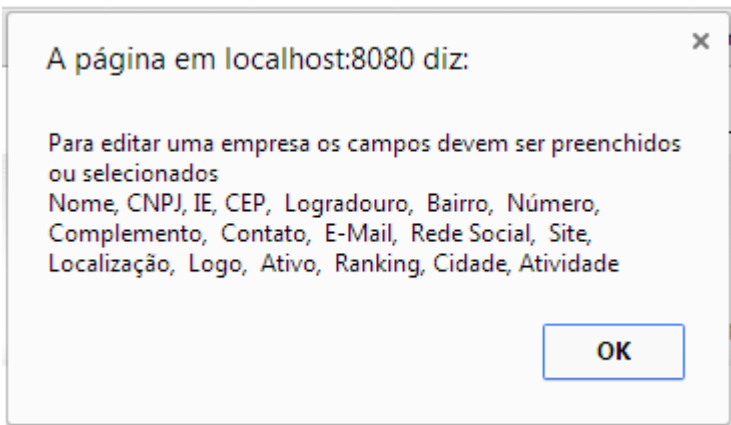

Data	Versão	Descrição	Autor
10/08/2013	1.0	Versão inicial	Maicon Alexander Illes Fang
11/12/2013	2.0	Versão secundaria	Maicon Alexander Illes Fang
25/02/2014	2.1	Ajustes e inserção de novos casos de teste.	Maicon Alexander Illes Fang

Log de Teste - UC001 – Manter Empresa

Cenário de Testes

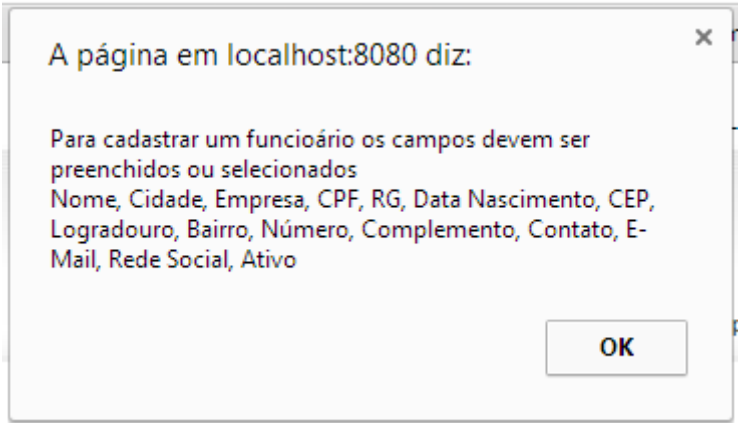
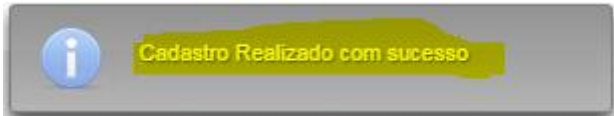

Use Case	UC001 – Cadastrar Empresa	Resultado Obtido
Número	Cenário	

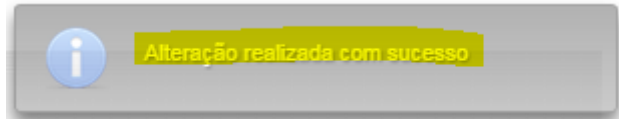
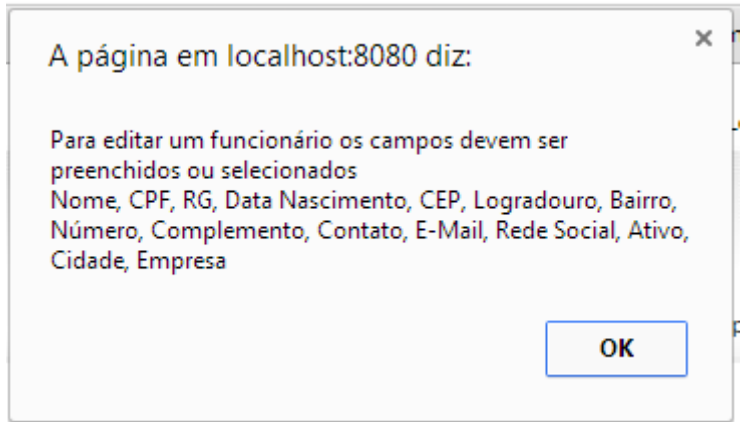
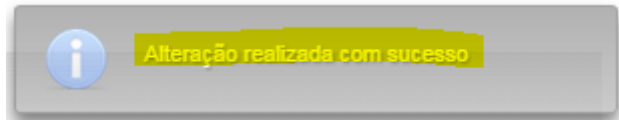
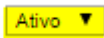
01	Cadastrar - Salvar registro sem preencher nenhum campo.	
02	Cadastrar - Validar cadastro com todos os campos preenchidos.	
03	Cadastrar - Validar CNPJ duplicado.	Não será implementado
04	Cadastrar - Validar campos com máximo de caracteres preenchidos.	
05	Cadastrar - Validar preenchimento de campos com caracteres inválidos.	Sistema não permite, pois caracteres inválidos são validados em javascript
06	Cadastrar - Verificar se campo "Ativo" é carregado com opção "Ativa"	<p>Ativo: </p> <pre><h:selectOneMenu id="ativo" styleClass="formulario_vObrigatorio" <f:selectItem itemValue="true" itemLabel="Ativa" /> <f:selectItem itemValue="false" itemLabel="Desativada" /> </h:selectOneMenu></pre>
Use Case	UC001 – Editar Empresa	
Número	Cenário	

07	Editar – Validar consistência das informações.	
08	Editar – Validar campo em branco	
09	Editar - Validar CNPJ duplicado.	Não será implementado
10	Editar - Validar campos com máximo de caracteres preenchidos.	
11	Editar - Validar preenchimento de campos com caracteres inválidos.	Sistema não permite, pois caracteres inválidos são validados em javascript
12	Editar - Verificar se campo “Ativo” é carregado com opção “Ativa”	Ativo: Ativa ▼

Log de Teste - UC002 – Manter Funcionário

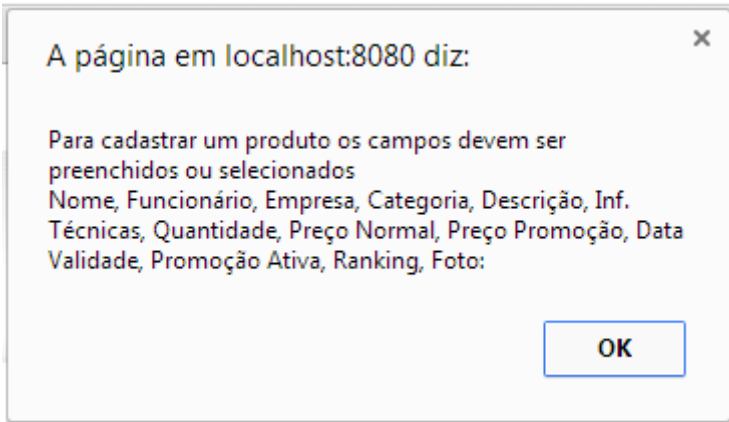
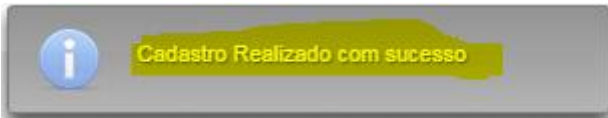
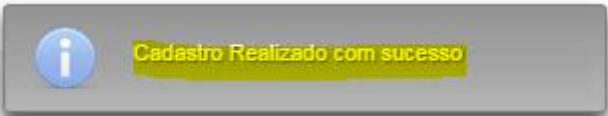

Cenário de Teste



Use Case	UC002 – Cadastrar Funcionário	Resultado Obtido
Número	Cenário	
01	Cadastrar - Salvar registro sem preencher nenhum campo.	
02	Cadastrar - Validar cadastro com todos os campos preenchidos.	
03	Cadastrar - Validar CPF duplicado.	Não será implementado
04	Cadastrar - Validar campos com máximo de caracteres preenchidos.	
05	Cadastrar - Validar preenchimento de campos com caracteres inválidos.	Sistema não permite, pois caracteres inválidos são validados em javascript
06	Cadastrar - Verificar se campo “Ativo” é carregado com opção “Ativo”	Ativo: Ativo ▼
Use Case	UC002 – Editar Funcionário	

Número	Cenário	
07	Editar – Validar consistência das informações.	
08	Editar – Validar campo em branco	
09	Editar - Validar CPF duplicado.	Não será implementado
10	Editar - Validar campos com máximo de caracteres preenchidos.	
11	Editar - Validar preenchimento de campos com caracteres inválidos.	Sistema não permite, pois caracteres inválidos são validados em javascript
12	Editar - Verificar se campo “Ativo” é carregado com opção “Ativo”	Ativo: 

Log de Teste - UC003 – Manter Produto

Casos de Teste

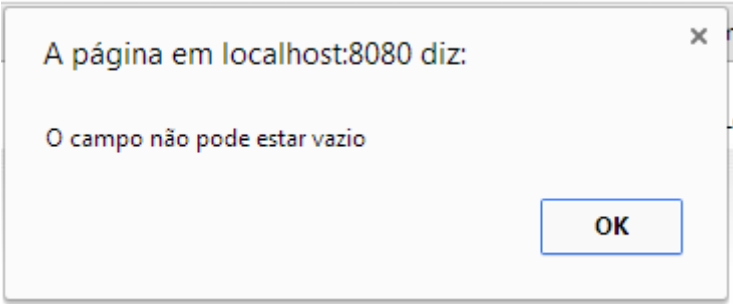
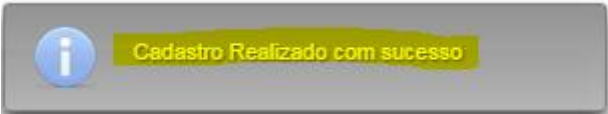
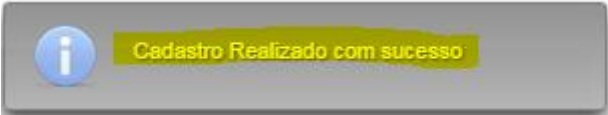
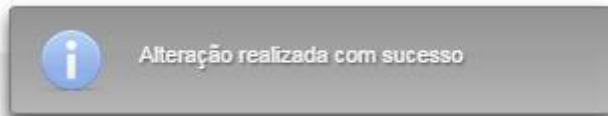
Use Case	UC003 – Cadastrar Produto	Resultado Obtido
Número	Cenário	
01	Cadastrar - Salvar registro sem preencher nenhum campo.	
02	Cadastrar - Validar cadastro com todos os campos preenchidos.	
04	Cadastrar - Validar campos com máximo de caracteres preenchidos.	
05	Cadastrar - Validar preenchimento de campos com caracteres inválidos.	Sistema não permite, pois caracteres inválidos são validados em javascript
06	Cadastrar - Verificar se campo "Ativo" é carregado com opção "Sim"	Promoção Ativa: 
Use Case	UC003 – Editar Produto	
Número	Cenário	

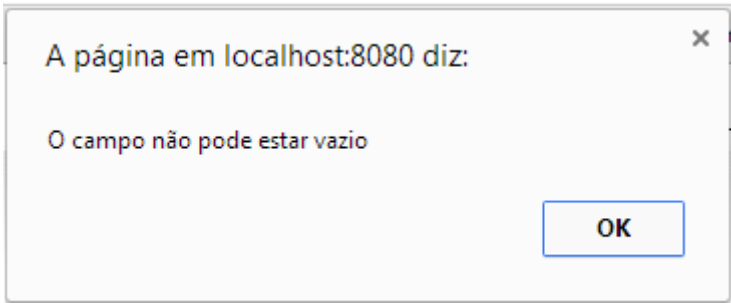
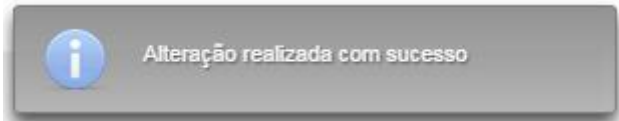
07	Editar – Validar consistência das informações.	
08	Editar – Validar campo em branco	
10	Editar - Validar campos com máximo de caracteres preenchidos.	
11	Editar - Validar preenchimento de campos com caracteres inválidos.	Sistema não permite, pois caracteres inválidos são validados em javascript
12	Editar - Verificar se campo “Ativo” é carregado com opção “Sim”	Promoção Ativa: 

Log de Teste - UC004 – Manter Categoria

Casos de Teste

Use Case	UC004 – Cadastrar Categoria	Resultado Obtido
----------	-----------------------------	------------------

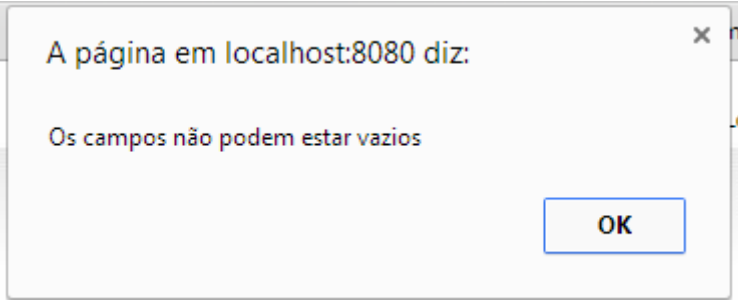



Número	Cenário	
01	Cadastrar - Salvar registro sem preencher nenhum campo.	
02	Cadastrar - Validar cadastro com todos os campos preenchidos.	
03	Cadastrar - Validar campos com máximo de caracteres preenchidos.	
04	Cadastrar - Validar preenchimento de campos com caracteres inválidos.	Sistema não permite, pois caracteres inválidos são validados em javascript
Use Case	UC004 – Editar Categoria	
Número	Cenário	
05	Editar – Validar alterações das informações.	

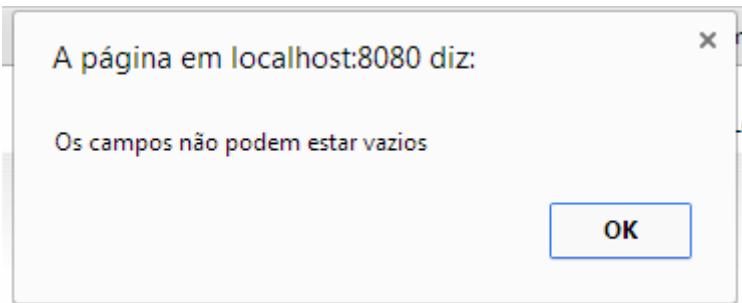
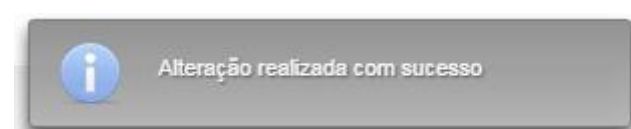
06	Editar – Validar campo em branco	
07	Editar - Validar campos com máximo de caracteres preenchidos.	
08	Editar - Validar preenchimento de campos com caracteres inválidos.	Sistema não permite, pois caracteres inválidos são validados em javascript

Log de Teste - UC005 – Manter Pais

Casos de Teste

Use Case	UC005 – Cadastrar Pais	Resultado Obtido
Número	Cenário	

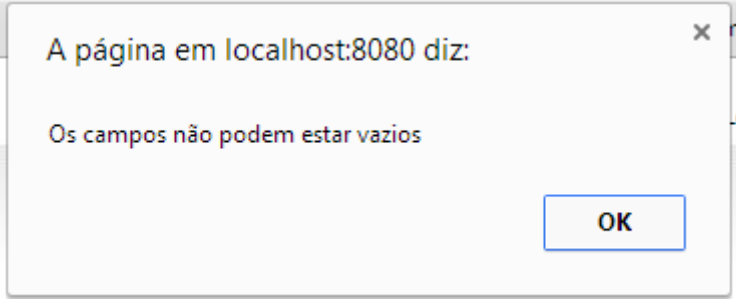
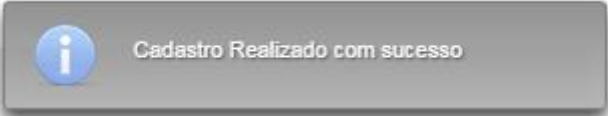
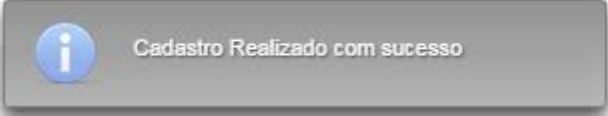

01	Cadastrar - Salvar registro sem preencher nenhum campo.	
02	Cadastrar - Validar cadastro com todos os campos preenchidos.	
03	Cadastrar - Validar campos com máximo de caracteres preenchidos.	
04	Cadastrar - Validar preenchimento de campos com caracteres inválidos.	Sistema não permite, pois caracteres inválidos são validados em javascript
Use Case	UC005 – Editar Pais	
Número	Cenário	
05	Editar – Validar alterações das informações.	

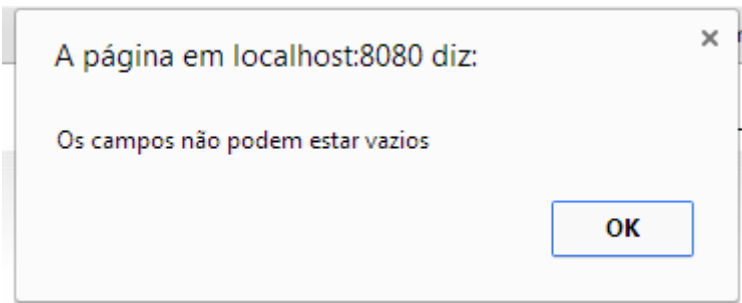
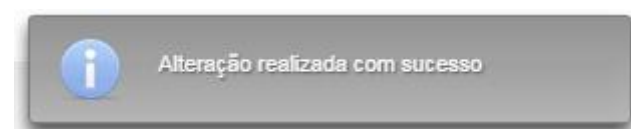
06	Editar – Validar campo em branco	
07	Editar - Validar campos com máximo de caracteres preenchidos.	
08	Editar - Validar preenchimento de campos com caracteres inválidos.	Sistema não permite, pois caracteres inválidos são validados em javascript

Log de Teste - UC006 – Manter Estado

Casos de Teste

Use Case	UC006 – Cadastrar Estado	Resultado Obtido
Número	Cenário	

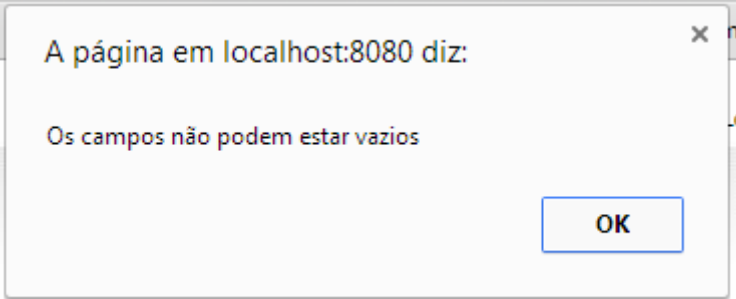



01	Cadastrar - Salvar registro sem preencher nenhum campo.	
02	Cadastrar - Validar cadastro com todos os campos preenchidos.	
03	Cadastrar - Validar campos com máximo de caracteres preenchidos.	
04	Cadastrar - Validar preenchimento de campos com caracteres inválidos.	Sistema não permite, pois caracteres inválidos são validados em javascript
Use Case	UC006 – Editar Estado	
Número	Cenário	
05	Editar – Validar alterações das informações.	

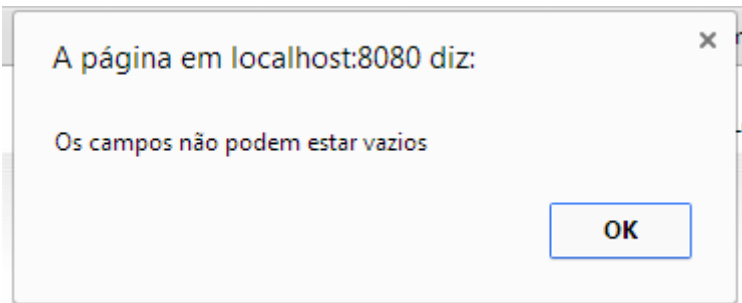
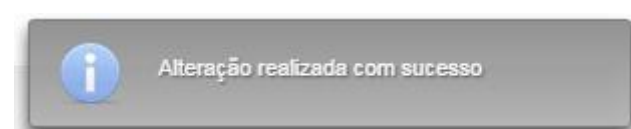
06	Editar – Validar campo em branco	
07	Editar - Validar campos com máximo de caracteres preenchidos.	
08	Editar - Validar preenchimento de campos com caracteres inválidos.	Sistema não permite, pois caracteres inválidos são validados em javascript

Log de Teste - UC007 – Manter Cidade

Casos de Teste

Use Case	UC007 – Cadastrar Cidade	Resultado Obtido
Número	Cenário	









01	Cadastrar - Salvar registro sem preencher nenhum campo.	
02	Cadastrar - Validar cadastro com todos os campos preenchidos.	
03	Cadastrar - Validar campos com máximo de caracteres preenchidos.	
04	Cadastrar - Validar preenchimento de campos com caracteres inválidos.	Sistema não permite, pois caracteres inválidos são validados em javascript
Use Case	UC007 – Editar Cidade	
Número	Cenário	
05	Editar – Validar alterações das informações.	





06	Editar – Validar campo em branco	
07	Editar - Validar campos com máximo de caracteres preenchidos.	
08	Editar - Validar preenchimento de campos com caracteres inválidos.	Sistema não permite, pois caracteres inválidos são validados em javascript

Log de Teste - UC008 – Manter Usuário

Casos de Teste

Use Case	UC008 – Cadastrar Usuário	Resultado Obtido
Número	Cenário	

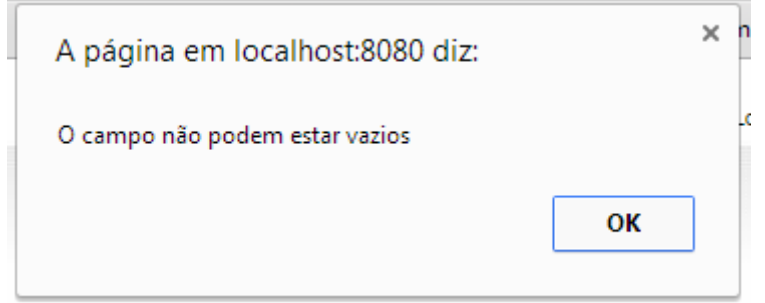

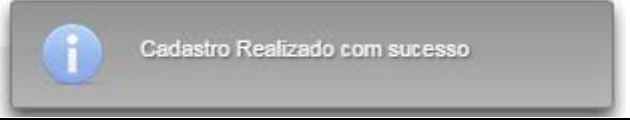

01	Cadastrar - Salvar registro sem preencher nenhum campo.	 O campo Matricula deve ser obrigatoriamente preenchido.  O campo Nome deve ser obrigatoriamente preenchido.  O campo Login deve ser obrigatoriamente preenchido.  O campo Email deve ser obrigatoriamente preenchido.  O campo Senha deve ser obrigatoriamente preenchido.  O campo confirme a senha deve ser obrigatoriamente preenchido.
02	Cadastrar - Validar cadastro com todos os campos preenchidos.	 Cadastro Realizado com sucesso
03	Cadastrar - Validar campos com máximo de caracteres preenchidos.	 Cadastro Realizado com sucesso
04	Cadastrar - Validar preenchimento de campos com caracteres inválidos.	Sistema não valida esse caso de teste

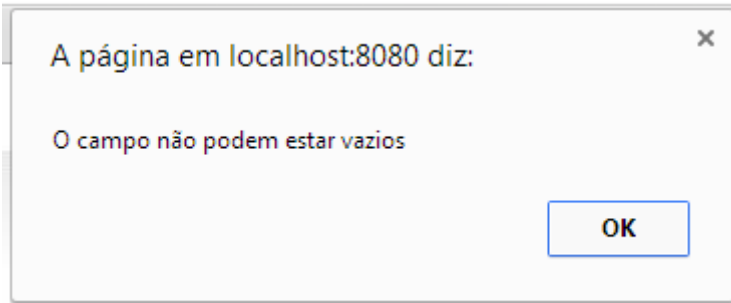
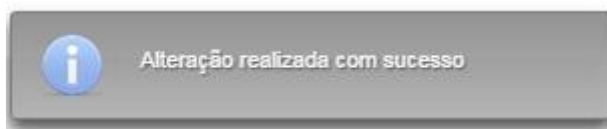
Use Case	UC008 – Editar Usuário	
Número	Cenário	
05	Editar – Validar alterações das informações.	
06	Editar – Validar campo em branco	 
07	Editar - Validar campos com máximo de caracteres preenchidos.	
08	Editar - Validar preenchimento de campos com caracteres inválidos.	Sistema não valida esse caso de teste

Log de Teste - UC009 – Manter Perfil

Casos de Teste

Use Case	UC009 – Cadastrar Perfil	Resultado Obtido
Número	Cenário	

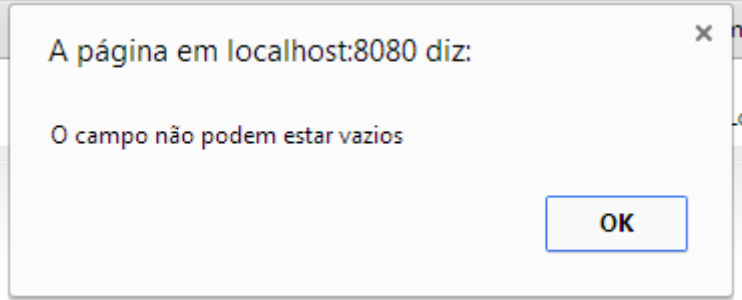
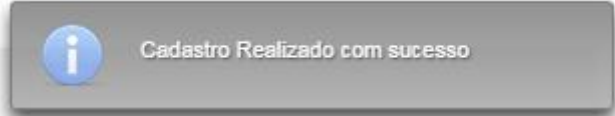
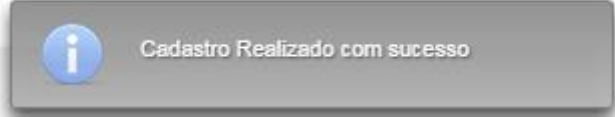
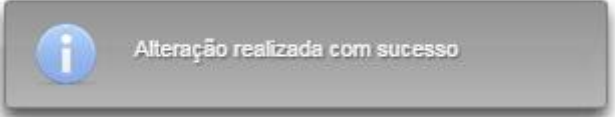
01	Cadastrar - Salvar registro sem preencher nenhum campo.	
02	Cadastrar - Validar cadastro com todos os campos preenchidos.	
03	Cadastrar - Validar campos com máximo de caracteres preenchidos.	
04	Cadastrar - Validar preenchimento de campos com caracteres inválidos.	Sistema não permite, pois caracteres inválidos são validados em javascript
Use Case	UC009 – Editar Perfil	
Número	Cenário	
05	Editar – Validar alterações das informações.	

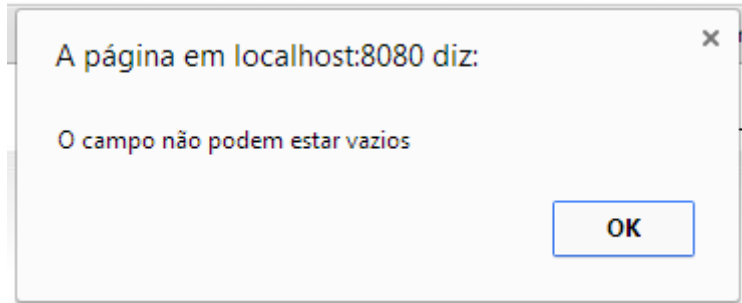
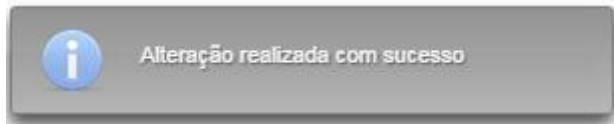
06	Editar – Validar campo em branco	
07	Editar - Validar campos com máximo de caracteres preenchidos.	
08	Editar - Validar preenchimento de campos com caracteres inválidos.	Sistema não permite, pois caracteres inválidos são validados em javascript

Log de Teste - UC010 – Manter Atividade

Casos de Teste

Use Case	UC010 – Cadastrar Atividade	Resultado Obtido
Número	Cenário	

01	Cadastrar - Salvar registro sem preencher nenhum campo.	
02	Cadastrar - Validar cadastro com todos os campos preenchidos.	
03	Cadastrar - Validar campos com máximo de caracteres preenchidos.	
04	Cadastrar - Validar preenchimento de campos com caracteres inválidos.	Sistema não permite, pois caracteres inválidos são validados em javascript
Use Case	UC010 – Editar Atividade	
Número	Cenário	
05	Editar – Validar alterações das informações.	

06	Editar – Validar campo em branco	
07	Editar - Validar campos com máximo de caracteres preenchidos.	
08	Editar - Validar preenchimento de campos com caracteres inválidos.	Sistema não permite, pois caracteres inválidos são validados em javascript

Log de Teste - UC011 – Manter Login

Casos de Teste

Use Case	UC011 – Login	Resultado Obtido
Número	Cenário	
01	Validar usuário desativado.	Não será testado, implementação futura
02	Validar usuário ativo.	Não será testado, implementação futura
03	Validar preenchimento de campos com caracteres inválidos.	Não foi possível se autenticar. Login ou senha inválido

Log de Teste - UC500 – Exibir empresa/s na aplicação Android

Casos de Teste

Use Case	UC500 – Empresa Android	Resultado Obtido
Número	Cenário	
01	Validar acesso a aplicação sem acesso à internet.	A aplicação informa que não foi possível
02	Validar acesso a aplicação com Webservice indisponível.	
03	Validar exibição das empresas sem nenhuma empresa cadastrada.	
04	Validar empresas desativadas no sistema.	

Log de Teste - UC501 – Exibir produto/s na aplicação Android

Casos de Teste

Use Case	UC501 – Produto Android	Descrição
Número	Cenário	Resultado Obtido
01	Validar acesso a aplicação sem acesso à internet.	N/A Não se aplica
02	Validar acesso a aplicação com Webservice indisponível.	
03	Validar exibição do produto sem nenhum produto cadastrado.	
04	Validar produtos desativadas no sistema.	

Log de Teste - UC502 – Exibir detalhe de um produto na aplicação Android

Casos de Teste

Use Case	UC502 – Detalhe do Produto Android	Resultado Obtido
Número	Cenário	
01	Validar layout da tela.	

APÊNDICE S - 19 - Solicitação de Mudança – Conserto de defeitos.

Solicitação de Mudança – Conserto de defeitos.

Histórico da Revisão

Data	Versão	Descrição	Autor
10/08/2013	1.0	Versão inicial	Maicon Alexander Illes Fang
11/12/2013	2.0	Versão secundaria	Maicon Alexander Illes Fang
25/02/2014	2.1	Ajustes e inserção de novos casos de teste.	Maicon Alexander Illes Fang

UC001 – Manter Empresa

Na apresentação do sistema para o usuário, foram levantados alguns itens encontrados na tela editar e gravar empresa que devem ser corrigidos.

1 - Sistema está permitindo duplicar número de CNPJ na base de dados.

2 - Sistema não solicita que seja selecionado a Cidade e Atividade ao gravar o registro com o restante dos campos preenchidos.

3 - Sistema não exibe dica de ajuda de campos e botões.

UC002 – Manter Funcionário

Na apresentação do sistema para o usuário, foram levantados alguns itens encontrados na tela editar e gravar funcionário que devem ser corrigidos.

1 - Sistema está permitindo duplicar número de CPF na base de dados.

2 - Sistema não solicita que seja selecionado a Cidade e Empresa ao gravar o registro com o restante dos campos preenchidos.

3 - Sistema não exibe dica de ajuda de campos e botões.

UC003 – Manter Produto

Na apresentação do sistema para o usuário, foram levantados alguns itens encontrados na tela editar e gravar produto que devem ser corrigidos.

1 - Sistema está permitindo duplicar nome de produto na base de dados.

2 - Sistema não solicita que seja selecionado a Funcionário, Empresa e Categoria ao gravar o registro com o restante dos campos preenchidos.

- 3 - Sistema não exibe dica de ajuda de campos e botões.
- 4 - Sistema deve carregar os campos Quantidade, Preço Normal e Preço Promoção sem nenhum valor.

UC004 – Manter Categoria

Na apresentação do sistema para o usuário, foram levantados alguns itens encontrados na tela editar e gravar categoria que devem ser corrigidos.

- 1 – Sistema não exibe dica de ajuda de campos e botões.
- 2 – Sistema permite duplicar nome de registro
- 3 – Na tela de consulta sistema não esta ordenando registro ao clicar no nome da coluna.

UC005 – Manter Pais

Na apresentação do sistema para o usuário, foram levantados alguns itens encontrados na tela editar e gravar pais que devem ser corrigidos.

- 1 – Sistema não exibe dica de ajuda de campos e botões.
- 2 – Sistema permite duplicar nome de registro
- 3 – Na tela de consulta sistema não esta ordenando registro ao clicar no nome da coluna

UC006 – Manter Estado

Na apresentação do sistema para o usuário, foram levantados alguns itens encontrados na tela editar e gravar estado que devem ser corrigidos.

- 1 – Sistema não exibe dica de ajuda de campos e botões.
- 2 – Sistema permite duplicar nome de registro
- 3 – Na tela de consulta sistema não esta ordenando registro ao clicar no nome da coluna
- 4 – Ao gravar ou editar algum registro sistema não informa ao usuário que o é necessario selecionar um pais.

UC007 – Manter Cidade

Na apresentação do sistema para o usuário, foram levantados alguns itens encontrados na tela editar e gravar uma cidade que devem ser corrigidos.

- 1 – Sistema não exibe dica de ajuda de campos e botões.
- 2 – Sistema permite duplicar nome de registro
- 3 – Na tela de consulta sistema não esta ordenando registro ao clicar no nome da coluna
- 4 – Ao gravar ou editar algum registro sistema não informa ao usuário que o é necessario selecionar um estado.

UC008 – Manter Usuário

Na apresentação do sistema para o usuário, foram levantados alguns itens encontrados na tela editar e gravar usuário que devem ser corrigidos.

- 1 – Sistema não exibe dica de ajuda de campos e botões.

- 2 – Sistema permite gravar duplicado matrícula, nome, login
- 3 – Na tela de consulta sistema não esta ordenando registro ao clicar no nome da coluna
- 4 – Ao gravar ou editar algum registro sistema não informa ao usuário que o é necessario selecionar um período.

UC009 – Manter Perfil

Na apresentação do sistema para o usuário, foram levantados alguns itens encontrados na tela editar e gravar perfil que devem ser corrigidos.

- 1 – Sistema não exibe dica de ajuda de campos e botões.
- 2 – Sistema permite duplicar nome de registro
- 3 – Na tela de consulta sistema não esta ordenando registro ao clicar no nome da coluna.

UC010 – Manter Atividade

Na apresentação do sistema para o usuário, foram levantados alguns itens encontrados na tela editar e gravar atividade que devem ser corrigidos.

- 1 – Sistema não exibe dica de ajuda de campos e botões.
- 2 – Sistema permite duplicar nome de registro
- 3 – Na tela de consulta sistema não esta ordenando registro ao clicar no nome da coluna.

UC011 – Manter Login

Na apresentação do sistema para o usuário, foram levantados alguns itens encontrados na tela editar e gravar login que devem ser corrigidos.

- 1 – Sistema não exibe dica de ajuda de campos e botões.
- 2 – Não foi implementado funcionalidade recuperar senha

APÊNDICE T - 20 - Plano de Implantação – Planejamento de como irá ocorrer a implantação do sistema no ambiente produtivo.

Planejamento de como irá ocorrer a implantação do sistema no ambiente produtivo.

Histórico da Revisão

Data	Versão	Descrição	Autor
10/08/2013	1.0	Versão inicial	Maicon Alexander Illes Fang
11/12/2013	2.0	Versão secundaria	Maicon Alexander Illes Fang
25/02/2014	2.1	Ajustes e inserção de novos casos de teste.	Maicon Alexander Illes Fang

Plano de Implementação

A baixo os passos para implementação do ambiente em produção

- 1 - Analizar a infraestrutura do ambiente de produção
- 2 - Criar usuários limitados para testes
- 3 - Analizar tráfego da rede
- 4 - Fazer testes de estresse, de carga e teste de vulnerabilidade do sistema
- 5 - Analisar resultados
- 6 - Apresentar um cenário confiável de uso

APÊNDICE U - 21 - Instalação – Passagem do sistema para o ambiente produtivo.

Instalação – Passagem do sistema para o ambiente produtivo

Histórico da Revisão

Data	Versão	Descrição	Autor
10/08/2013	1.0	Versão inicial	Maicon Alexander Illes Fang

11/12/2013	2.0	Versão secundaria	Maicon Alexander Illes Fang
25/02/2014	2.1	Ajustes e inserção de novos casos de teste.	Maicon Alexander Illes Fang

INSTALACAO

Alguns passos devem ser feitos antes de ter acesso as funcionalidades do sistema, abaixo são listados os itens.

- 1 – Ter o Apache Tomcat 7 instalado e configurado na máquina.
- 2 – Ter o MySQL instalado e configurado no computador.
- 3 – Ter o Chrome ou Firefox instalado no computador.
- 4 – Ter a aplicação web configurada no computador.
- 5 – Sistema web e dispositivo móvel terem acesso a internet
- 6 – Ter a aplicação instalada no dispositivo móvel